DDS BASED SMART GRID DATA INTEROPERABILITY AND PERFORMANCE MEASUREMENT

BY

HASSAN ALI

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER NETWORKS

DECEMBER 2016

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **HASSAN ALI** under the direction of his thesis adviser and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN COMPUTER NETWORKS.**

Thesis Committee

Dr. Basem AL-Madani (Adviser)

Dr. Marwan H. Abu-Amara (Member)

Dr. Abdulaziz Y. Barnawi (Member)

Dr. Ahmad Almulhem
Department Chairman

Dr. Salam A. Zummo
Dean of Graduate Studies

22/1/17

Date

عمادة الدراسات العليا
DEANSHIP OF GRADUATE STUDIES

Dedication

To my Parents for their LOVE

ACKNOWLEDGMENTS

In the name of Allah, the most Gracious and most Merciful

All praise to Allah Almighty Who allowed me to be a part of KFUPM community. It was a great opportunity to study in such prestigious institute. I thank to KFUPM for providing almost everything to support the research work of Masters thesis. All appreciation to my adviser Dr. Basem Al-Madani, who helped me, guided me, encouraged me and teach me so many things during this journey. I also want to thank my thesis committee members, Dr. Marwan H. Abu-Amara and Dr. Abdulaziz Yagoub Barnawi for their help and support. I appreciate and would like to thank Real Time Innovations (RTI) Inc. for their excellent products with free licensing, training videos and tutorials that are readily available for educational purposes and research work.

In the end I want to express my gratitude towards all of my family members for their prayers and support in all ups and downs. I would like to thank all of my KFUPM colleagues and Pakistani community with whom I passed such a memorable time.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

DDS	Data Distribution Service
QoS	Quality of Services
RTI	Real-Time Innovations
ANSI	American National Standards Institute
SCADA	Supervisory Control and Data Acquisition
DR	Demand Response
WAMPAC	Wide Area Monitoring Protection and Control
RTU	Remote Terminal Unit
HEM	Home Energy Management
AMI	Advanced Metering Infrastructure
OS	Operating System
IED	Intelligent Electronic Device
PMU	Phasor Measurement Unit

MP2P	Multi-Point to Point
P2MP	Point to Multi-Point
P2P	Point to point
WSN	Wireless Sensor Network
RTPS	Real Time Publish Subscribe
IDL	Interface Descriptive Language
DCPS	Data Centric Publish Subscribe
API	Application Programming Interface
OMG	Object Management Group
NEMA	National Electrical Manufacturers Association
RTT	Round Trip Time
PV	Photo Voltaic
SOAP	Simple Object Protocol
CORBA	Common Object Request Broker Architecture

THESIS ABSTRACT

NAME:	Hassan Ali
TITLE OF STUDY:	DDS based Smart Grid data interoperability and performance measurement
MAJOR FIELD:	Computer Networks
DATE OF DEGREE:	December, 2016

Today's power grid has so many challenges in terms of centralized power generation, limited flow of information, limited support for power distribution, poor management of peak loads and power disruptions. Due to these limitations several organizations are working on Smart grid. As Smart grid consists of numerous kind of heterogeneous devices which increase the complexity and inefficiency. To cope with heterogeneity and provide interoperability for communication of these devices, middleware is considered to be the best approach. There are so many middlewares that have been proposed so far but DDS middleware provides high level of reliability and efficiency by addressing more performance metrics and several QoS policies especially in real time and mission critical applications. DDS fulfills almost all of the communication requirements of Smart grid due to rich set of QoS

policies. We have deployed DDS in generation side of Smart grid in which energy renewables data is published to control and monitoring station. Similarly we have considered Smart grid standard as ANSI C12.19 to deploy DDS in transmission and consumption sides as well. Data structures are obtained to form topics over DDS RTI Connext to establish communication and conduct experimental study to analyze interoperability and other performance metrics. In the end we have compared publish subscribe performance of DDS with SocketIO based client server application in which DDS is proved to be a much better solution for Smart grid data interoperability and high performance efficiency.

CHAPTER 1

INTRODUCTION

The traditional power grid deals only with the data that it has supplied but obtain no data in response for the delivery of energy and power. These responses are much necessary to analyze the suitable tariffs for the customer and day to day requirements for the shifting and changing of electricity. That is why to save the resources, get the feedback of user data and to minimize the electricity consumption, several efforts and attempts are being made by the researchers to shift from regular Power grid to Smart grid. Smart grid is modern power grid that uses the digital and analogue communication system to collect the information and then it responds accordingly. In this way, it gathers the information from consumer and suppliers in an automated way thus ensures the reliability, efficiency, economics, scalability and sustainability of the production and distribution of electricity [1]. A rough sketch of Smart grid is shown in Figure 1.1. For this advancement several modifications of existing elements or new elements are required to upgrade the entire system. Among these elements, need of middleware is much essential

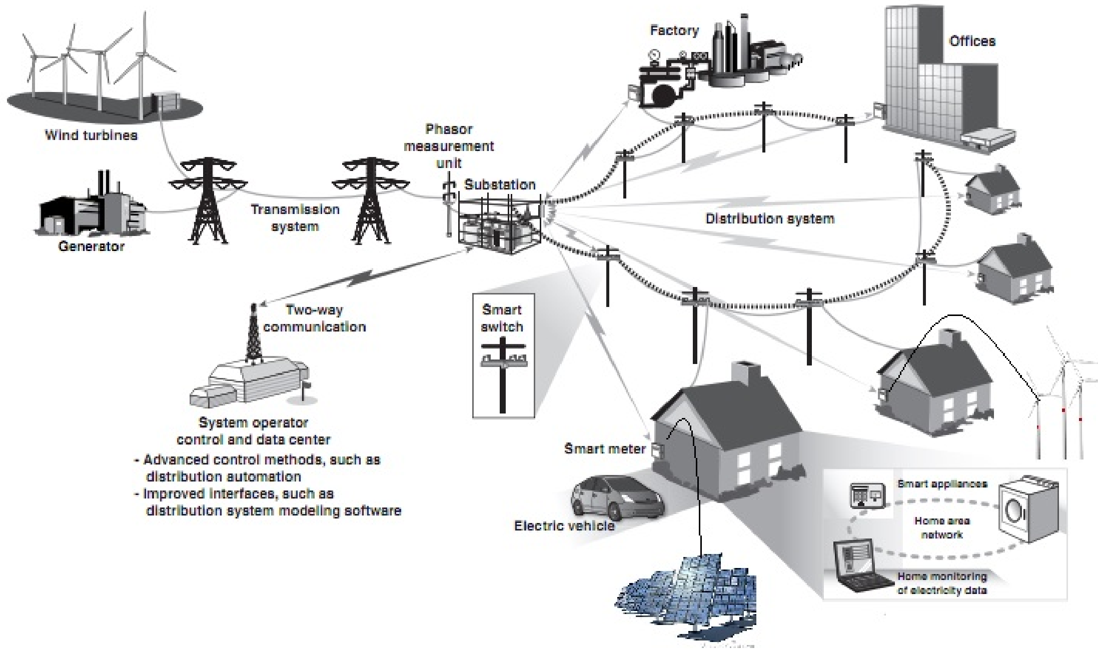

Figure 1.1: Smart Grid Overview

to establish the communication among several already used devices like embedded systems for power transmission [2]. Smart grid depends entirely upon the communication of numerous state of the art devices. These devices are heterogeneous in nature so as to provide interoperability, establish reliable communication among these devices and to provide Quality of Services (QoSs), the middleware architecture is the best solution so far. Various middlewares for Smart grid communication have been proposed by the researchers along with comparative studies to analyze which middleware addresses which kind of performance matrices. As Data Distribution Service (DDS) provides almost every kind of performance metrics so we considered DDS middleware to conduct experimental study over Smart grid. DDS communication model is based on publish subscribe architecture so we considered Smart grid devices as publisher and subscriber applications to communicate with each other. ANSI C12.19 standard and energy renewables are used to get data structure to build topics to establish communication among these devices by setting certain QoS contracts provided by DDS. RTI Connext is used to perform publish subscribe communication and to conduct several experimental analysis that include interoperability, throughput, jitter, latency etc.

1.1 Limitations of existing power grids

Currently, central power plant is responsible for electricity generation for power grid. Electricity then transferred to end users over transmission lines. From generation sides, voltages have to be stepped up for power transmission and then stepped down at power grids, it may stepped down again for the utilization of end user customers. In this phenomenon of transferring power from far locations to end users, a lot of line losses occur which ultimately causes rapid increase in costs as well as wastage of resources. Moreover, current power grids do not accommodate such latest technological trends so that they may become a part of advanced economy, energy generation and conservation. There are some major issues and restraints in current power grids that are discussed below [3] [4] [5].

1.1.1 Centralized generation

Power grids that are currently being utilized, have centralized generation that means it can only generate power in one location or a few and has to deliver it in vast areas to meet consumer needs. Centralized generation causes line losses, difficult maintenance due to huge capacity and no backup for power generation in case of shutdown. Centralized generation must be converted to, or accommodated with distributed generation with energy renewables such as solar rooftops, bioenergy and wind turbines. Consumers must also have the capability to send excessive electricity to power grid which is called net metering.

1.1.2 Limited flow of information

Bidirectional flow of information among consumer and power grid does not exists to communicate effectively. In this way all of the information from consumer side mostly, cannot reach to power grid management which is much essential for robustness and efficiency.

1.1.3 Delivery system restraints

Now a days, engineers are using Supervisory Control and Data Acquisition System (SCADA) for electricity transmission at an advanced level but it consumes a lot of bandwidth. This huge bandwidth consumption may result in low data rate which results in low efficiency and delay to respond for an alarm and state change for feedback systems.

1.1.4 Poor supply demand management

Due to dynamic behavior of delivery costs and electricity needs, it is necessary to have efficient demand response, supply and management for power transmission.

Current power grids have poor efficiency, advance supply mechanisms and management for peak loads, therefore, they suffer with high cost, poor emissions and low power delivery.

1.1.5 Power interruptions, blackouts and outages

Due to lack of back up for power generation and transmission, consumers and end users suffer a lot with power outages, interruptions and blackouts. Mostly under developed countries facing this problem due to mismanagement of power systems [3]. USA alone suffers with the cost of about 150 billion dollars per year for such problems [3]. Industries are the biggest victims that contribute to economy heavily which is a huge country level set back [5].

1.2 Smart Grid

As regular power grid does not accomplish all the demands of required advancements, state of the art technologies and communication infrastructure related to power market so we need a shift from regular grid to Smart grid. Smart grid eyes on to various state of the art technologies, new policies and customer solutions. Smart grid has a potential to address all the challenges in power system from generation to distribution and consumption ensuring reliability, sustainability, efficiency, demand response support and load balancing adjustments. It accommodates several new concepts that includes distributed generation, meet demand and response based on feedback system, new taxonomies and standardizations. Moreover, a lot of research work is being done on Smart grid and its advancements. The definition of Smart grid according to US department of Energy is: [6]. "A smart grid uses digital technology to improve reliability, security, and efficiency (both economic and energy) of the electrical system from large generation, through the delivery systems to electricity consumers and a growing number of distributed generation and storage resources."

In Table 1.1 several major differences between Smart grid and regular grid have been identified through literature reviews [3] [4] [7] [8].

Table 1.1: Regular grid Vs Smart Grid

Regular Power Grid	**Smart Grid**
It has Electromechanical system	It contains digital system
Operates with reliability estimations	Operates with reliability predictions
Manual monitoring and maintenance	Automatic monitoring and maintenance
Fixed monitoring, control and protection systems	It uses adaptive and WAMPAC protection systems
Centralized generation	Distributed generation
Solid state build relays	Microprocessor build relays
Manual recovery	Automatic recovery and fault tolerance capability
On site device monitoring	Remote device monitoring
Full duplex (locally) and half duplex communication	Integrated and global full duplex communication
Limited control system	Ubiquitous control system
Limited choices for customers	Broad range of choices for customers
Minor integration with limited market for new products and services	Well Integrated and vast markets
No importance for power quality	Provides value to power quality for digital economy
Inefficient management of operational data and assets	Effective data management and assets
Limited volume of sensors	Large volume of sensors
No option for energy storage	Energy storage systems available

Smart grid is surely a replacement of regular power grid but it faces some challenges in deployment. These challenges mostly deal with management, security and reliability. Smart grid at country level needs standardization of technologies to be accommodated in it. Because Smart grid integrates new high-tech abstract models and operations for utilities that are made by scientists and engineers across the globe and they keep advancing and increasing, so the policy makers, consumers and regulators are somehow not clear of latest concepts and technologies. Standardization is also necessary for the efficient estimation of cost and benefit analysis. As Smart grid consists of huge communication infrastructure, that will be detailed later on, so there is another challenge of information security. Due to state of the art networking and communication technologies Smart grid is vulnerable to cyber-attacks. So the Smart grid must be protected and aware for such threats and security concerns.

1.3 Communication requirements of Smart grid

Smart grid concept deals with the integration of advanced communication infrastructure and information to achieve an efficient and high level use of distributed, new technologies and heterogeneous environment. Smart grid contains a communication network that will obtain data from smart, seamless and flexible management of hardware devices for efficient energy utilization [9]. Data of different nature can be forwarded efficiently in Smart grid communication infrastructure by fulfilling its communication requirements. Hence, an interoperable, flexible, integrated and two way communication is required to fulfill the basic need of Smart grid communication infrastructure with sufficient bandwidth and low latencies [10]. Furthermore, there must be a robust system security with advanced controls that provides reliability, stability and avoids cyber-attacks [11]. Smart grid requires such integrated energy infrastructure that fulfills the advanced requirements of electricity production, consumption and delivery. It must have such communication system that is useful for distributed and automated monitoring, optimization and management functions. The key communication priorities are:

- To guarantee real time collection and management of data from large number of data resources that is different are nature.
- To deal with different communication services, for instance unicast or multicast that are required by power application controls in power system.

Moreover there are some issues related to the communication infrastructure of Smart grid such as design of communication network, use of suitable technologies to link several devices, use of suitable network topologies and protocols to fulfill the Smart grid communication requirements [12].

1.3.1 Distributed Communication

The smart grid needs to be equipped with decentralized and distributed communication monitoring and control systems unlike the traditional grids. The shift from vertical communication to horizontal communication will require more flexible communication protocols that will allow data exchange between Remote Terminal Units (RTUs) and controllers horizontally [13].

1.3.2 Latency

Some of the applications such as Home Energy Management (HEMs) and Advanced Metering Infrastructure (AMI) may allow a little latency whereas applications like distributed automation and wide area awareness systems are mission critical applications and do not allow any latency. Control signals of power system, online sensors and meter reading in smart grid require real time data communication [14]. Mostly, communication interactions in Smart grid are time constraint and take place in real time. Latency requirements related to various network applications are divided into six categories that are mentioned below with their maximum response time lines [15] [16].

1. *Protection:* 1 to 10 milliseconds
2. *Control:* 100 milliseconds
3. *Monitoring and reporting:* 1 second
4. *Metering and sensing:* 12 to 20 milliseconds
5. *IEDs to data aggregator:* 4 milliseconds
6. *Data aggregators and Utility control centers:* 6 to 12 milliseconds

1.3.3 Reliability

A simple and reliable data transfer according to the specific communication requirements exists under the scope of reliability. Various applications in Smart grid such as distributed automation, requires high reliability so the communication entities must be always reliable in order to maintain the persistence of communication [17]. Some of the functions in Smart grid require higher reliability even up to 99.9 % that means it can only allow one second outage annually [18]. Providing reliability in Smart grids has been more challenging because of the certain factors [19] that are mentioned below:

- Increase in energy demand which causes more entities to be connected in Smart grid.
- Increase of resource utilization which is to be monitored and controlled.
- Limited rights and lack of investment.
- Operating of grid at maximum capacity level.
- Long distances energy transformation.
- Diversity, uncertainty and grid congestion.

1.3.4 Data Rate

More often the data rates in a single communication channel become serious concern when there are a large amount of Intelligent Electronic Devices (IEDs) need to be connected in utility core of Smart grid. Different applications in Smart grid may require different data rates [17]. For instance, data transmission of audios and videos that are used for device monitoring purposes, use Wide Area Networks of LAN and WiFi that require high data rates between 11 to 110 Mbps, to obtain precise and reliable communication whereas, AMI and distributed automation that use ZigBee or WiMax may require low data rates between 55 to 70 Mbps [20]. However, smooth flow of data must always be ensured for high performance.

1.3.5 Bandwidth

The communication infrastructure must be able to handle a large amount of messages without any delay or latency in a scalable system of Smart grid [14]. Network bandwidth must be upgradable according to the increasing demand of incoming intelligent devices. Smart meters and other meter reading sensors in Smart grid usually require moderate kind of bandwidth i.e. 300 kbps [12]. However the growing demand of bandwidth in Phasor Measurment Units (PMUs) and IEDs can possibly be between 10 kbps to 100 kbps [18].

1.3.6 Throughput

Throughput is known as the rate on which messages can be transferred successfully over a communication channel. The application related to communication system such as Demand Response (DR) and AMI estimated to demand a throughput between 3 to 10 Mbps [21].

1.3.7 Interoperability

Interoperability deals with the heterogeneous components of Smart grid to be work together in order to collectively perform different functions, effective exchange of information and to offer compatibility. It needs two way communication, efficient collaboration and integration among several heterogeneous elements to get connected together. But there is a deficiency of interoperability standards due to which it may become difficult to set up communication networks in Smart grid. To provide complete set of networking solutions, interoperability must be ensured at each division in Smart grid. [22].

1.3.8 Flexibility

It is considered as a multi-faceted notion or idea in context of Smart grid communication infrastructure. For instance, flexibility provide the support for varied nature of services in Smart grid such as heterogeneous networks and operating systems etc. that have various timeliness and reliability requirements [12], whereas flexibility also supports various kinds of communication models such as Multi-Point to Point (MP2P) or many to one that require collection of status information from various sensors periodically. Point to MultiPoint (P2MP) or one to many communications and group based communications that are used to deliver configuration instructions and commands to electrical devices are equally useful as MP2P [23] [24]. As a whole, networking protocols and technologies with high quality of adaptability and flexibility are required to satisfy the need of various applications to communicate with each other within same communication infrastructure.

1.3.9 Scalability

The communication infrastructure in smart grid need to entertain more incoming devices such as data collectors, sensor nodes, smart meters, and renewable energy resources. Thus providing scalability along with integration of authentic protocols, smart web services and end user interaction devices with advanced functionalities [14]. There are two kinds of scalabilities that are required in smart grid:

1. *Geographic Scalability:* Broad configuration and sizes of network deployment in vast areas.

2. *Load Scalability:* To handle additional volume of service requests and data traffic in communication system.

But here we are only concerned with load scalability to handle large communication infrastructure. Distributed communication architectures [25] have been developed that support internet services in a scalable aspect that can be deployed in Smart grid but a number of problems will arise such as limited computing, storage and communication abilities.

1.3.10 Security

End to end security is much critical particularly for mission critical applications to avoid vulnerabilities of major Smart grid devices [26]. Vulnerabilities may also allow an adversary to infiltrate into the Smart grid network and change load conditions or acquire control over software to weaken the functionality of Smart grid [14] specially using Wireless Sensor Networks (WSNs) in AMI. Denial of Service attack and its avoiding technique is identified in IEEE 802.15.4 Smart grid WSNs [27]. The consumer and metering network privacy is very crucial to gain the trust of people. A research has been conducted in this area in which user will have to reassure that their data is secure enough and reach specific location within Smart grid [28].

1.3.11 Complexity

A large amount of devices and components that may be managed remotely are interconnected with each other thus need so many requirements to be fulfilled which makes Smart grid a complex system. Thus it is a big challenge in modeling, designing and analyzing a communication infrastructure that meets these requirements. The resilience and graceful degradation of uncertainties and inconsistencies must be provided within the communication infrastructure and control system of Smart grid. Emerging behavioral simulations, uncertainties, use of complicated numerical tools, large scale analysis and so many other aspect are causes of complexity in smart grid [14].

1.3.12 Efficiency

The efficiency in terms of maximum output that can be obtained in communication system has challenges that are being increased due to modern communication infrastructure and technologies in Smart grid that deliver sufficient control and monitoring capabilities [14]. A high degree of global analysis along with intelligent and fast responses is required from the communication infrastructure in order to prevent rapid unfavorable events [29]. It needs to address following challenges:

- **Network Technology:** Wired and wireless network technologies may be deployed as per conditions of the system and surrounding environment. Distributed architecture must be implemented along with well-defined network protocols that provide cyber security, process control and exchange of data [30].

- **Improved Computing Capabilities:** Reliable analysis based on mission critical data to back operator's decision, a temporally and geographically hierarchy of communication and accommodation of intelligent agents must be given secure and fail proof communication systems in order to enhance their efficiency [30].

- **Secure and integrated communication:** Two way communications among customers and operators and flexible network configurations to satisfy fail proof automation and monitoring can be delivered through open standard pervasive and highly distributed communications [30].

- **IEDs communication:** Autonomous control processes, equipment repair, Remedial Action Scheme (RAS) and System Protection Scheme (SPS), are among those actions and operations that are very essential to identify and prevent faults, bad data and constraints in order to provide adaptive and intelligent communication [31].

- **Swift Controls:** The management of power and voltage flow at generation, transmission and distribution require swift automated control processes that are enabled by communication infrastructure based on data of power electronics.

1.3.13 Robustness and Availability

Robust and swift control devices, modern communication protocols, advance data communication and IEDs of entire grid from utility to customers will ensure the robustness significantly [32]. Whereas availability is required in communication structure that may be achieved through adaptive communication technologies. Wireless technologies are the preferred ones because they offer low cost installations with constrained security and bandwidth to be deployed in Smart grid on large scale, whereas wired technologies provide high degree of security, reliability, accuracy and capacity but are expensive ones due to its deployment and maintenance costs [33].

1.3.14 Predictability and intelligence

System must be able to perform normal and Emergency mode of operations. It must have ability to stay connected even in dynamic behavior of voltage and frequency range changes. It is essential to have the ability to predict the overall behavior that includes frequency behavior, steady state voltage behavior, transient voltage behavior, state current behavior, transient current behavior and behavior of other equipment and devices in Smart grid [34].

1.4 Need of Middleware technology in Smart grid

Middleware is a software that is much useful in integration of several technologies and applications in distributed systems. Middleware addresses the challenges of complex coordination among devices, its location may be seen in Figure 1.2 [35].

Middleware can be deployed at generation, transmission and consumption sides and helps to fulfill broad range of communication requirements [36]. Smart grid infrastructure is extremely dynamic because it offers certain benefits to consumers to change the demand of power so that it can manage generation resources efficiently. This dynamic behavior force Smart grid to deploy new technologies and make some adjustments. There are numerous applications to upgrade services of Smart grid [37] [38] but these applications need exact estimation of states of Smart grid. Abundant data is needed from millions of sensors to broadcast Smart

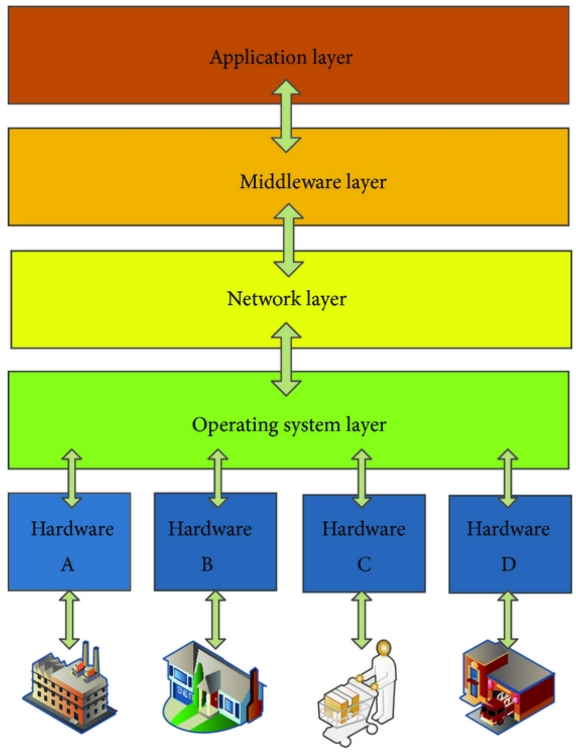

Figure 1.2: Location of Middleware Layer

grid states which must be provided to relevant applications and devices in real time with various formats. Transmission of data from Smart meters and PMUs kind of sensors that are deployed in Smart grid must overcome real time mission critical constraints. There is no such software to handle, manage, store, deliver and retrieve this kind of data efficiently and reliably. Middleware is a need of the hour in Smart grid not only for reasons described above but also for monitoring and control of heterogeneous entities at large scale. Middleware provides an efficient and reliable communication through which estimations of power states and consumption can be made easily, that also ensures sustainability and stability [39]. Middleware serves as a key role in real time mission critical Smart grid communication infrastructure that provides interoperability along with other solutions [40].

1.4.1 Interoperability

Interoperability at middleware level deals with data communication. There are two types of interoperabilities [41] that need to be addressed in Smart grid.

1. Syntactic Interoperability: It deals with individual elements data structure that must be recognized during transmission of messages within the system.

2. Network Interoperability: it deals with the transmission of messages in different networks.

1.4.2 Integration of applications and technologies

Implementation of abundant and diverse applications such as DR, distributed generation and smart meters etc. with legacy and propriety protocols needs integration because they have to work together within an intelligent framework. Various devices with different functions must be integrated with one another to handle information over leading communication and control technologies to fulfill user demands. Thus a middleware is required to act as a gateway for integration of these applications and devices in distributed environment [42].

1.4.3 Innovative Transformations

Smart grid is getting evolved day by day thus new technologies and advanced applications are being introduced continuously. In order to make Smart grid more powerful to accommodate current and future innovations, it must have the functionality of scalability at each level. At communication level innovative technologies always require reliable, secure and flexible range of IP addresses that may deliver real time data transactions and updates. These requirements enforce Smart grid infrastructure to have a middleware that can ensure scalability in almost every aspect [43].

1.4.4 Need of bridge between consumer and system

Smart grid deals with the two way communication among system and end users. The two way communication required a bridge to connect both endpoints and to overcome the gaps that may occur in between. A middleware is needed to monitor and control various equipment among various sites of Smart grid such as generation, transmission and substation to collect accurate data. Middleware ensures efficient resource handling and data transformation by acting as bridge between system, devices and end users [39] [44].

1.4.5 Abstraction

Smart grid deals with much complexity in its various operations, as millions of devices keep sending data to one another to accomplish certain tasks. By hiding these complexities from operators, developers, devices and users, the system becomes much desirable and an excellent one. Thus middleware is needed to provide such abstraction in modeling and interconnection of distributed systems between operating system and application programs. Moreover, middleware can provide various level of abstractions [45] that are described below:

- Middleware provides network centric abstraction in high level building blocks that includes objects, distributed tuples and variables.
- It provides abstraction of low level networking by hiding unnecessary details from programmers thus ensuring user friendly environment that results in robustness and high performance.
- It also keeps user unaware with the deep structure of power grid such as embedded systems, low level devices and measuring components.

1.4.6 Information and data management

Data and Information from heterogeneous devices in Smart grid may be illustrated in different formats such as little endian or big endian or any other. Thus wrong interpretation of data will be a disaster in such constrained environment. Moreover, real time data may also be used to actuate some hardware devices [46] which is much sensitive in nature. To handle such precise and string information accurately there should be an agreement within Smart grid so that efficient and desired outcomes can be obtained. Middleware is such technology that can manage such sensitive information elegantly by ensuring autonomous data transmission [47].

1.4.7 Monitoring, Control and Measurement analysis

Smart grid infrastructure consists of monitoring and control operations of applications at almost every level. For example at system level operator or system administrator needs so many readings and measurements of various devices to monitor and control so that timely decisions may be taken. At user level, it needs to monitor and control electricity consumption by each appliance, electricity generation from more than one source (if have), batteries and so many other things. On the other hand secure and reliable communication may achieve through Wide Area Monitoring Protection and Control (WAMPAC) system [48]. But using so many systems for monitoring and control is complex and challenging that is why there is a need of a middleware service that can manage all of these advanced monitoring and control systems in Smart grid. Moreover, Smart grid also includes WSNs that are used to take readings from several places and devices. These WSNs are much important in monitoring, control and decision making process because they deliver real time data but incur problems such as time constraint data transfer which affect quality of service in Smart grid. Quality of Service is an essential task that can be served through middleware [49].

1.5 Research objectives

The project eyes on thorough and comprehensive understanding of DDS based middleware which is a most advanced system for heterogeneous and distributed systems. The purpose of this work is comparative analysis of various performance measures over various network technologies. Project research objectives includes:

- Identify, understand and format data structures from standards of smart grid such as ANSI C12.19 to build topics in DDS to establish communication between publisher and subscriber at distribution and consumer side. Also analyze performance by changing number of publishers, subscribers, packet sizes and QoSs. We mainly focus on Latency, Throughput and Bandwidth communication requirements of Smart grid in our study and experimental work.

- Analyze data interoperability provided by DDS having publishers and subscribers in Smart grid with different OS and architectures from each other.

- Verification and mapping of Smart grid communication requirements with the Quality of Services provided by DDS.

- Integration of energy renewables and other heterogeneous devices in Smart grid and analysis of performance metrics such as jitter, latency, throughput and bandwidth utilization in several different scenarios for instance by varying number of publishers and subscribers and message sizes among LAN, WIFI.

- Propose an integration and interoperability platform to develop information sharing Smart grid applications to enhance performance and efficiency to uphold Smart grid real time constraints. The platform based on DDS will be analyzed as a part of Smart grid communication infrastructure that will be able to synchronize new and existing applications that will allow dynamic transactions among these applications.

1.6 Research methodology

To analyze interoperability in various applications of Smart grid, we will implement DDS based publish subscribe communication system on various operating systems and architectures. For this we need to build several topics based on data structures. At one side we have considered ANSI C12.19 standard (which provides data tables for smart grid end devices) and on the other hand we use energy renewables data control parameters to get data structures to build topics. Data writer of publisher will be subjected to write data in topics based on certain specified Quality of Service parameters and data reader of subscriber will be subjected to read data based on same Quality of Services from topic. Communication will be established between both heterogeneous applications and we will analyze the data interoperability. Performance will be measured by comparing DDS with other real time communication model based on latency, throughput and jitter etc.

1.7 Thesis breakdown

Having presented the rationale for the research problem and laid down the foundation of the area the problem belongs to in the first chapter. The second chapter will briefly explain the Real time publish subscribe middleware and fundamental concepts used in this work. Chapter 3 will mention some of the related works already done in the field of study to show the big picture and assert the significance and relevance of our work in comparison with existing framework. In chapter 4, we will present implementation of ANSI C12.19 Smart grid standard over DDS based RTI Connext. Chapter 5 includes integration of several various energy renewables in Smart grid using DDS middleware. Finally chapter 6 will conclude the finding and hint future vistas for further research.

CHAPTER 2

DATA DISTRIBUTION SERVICE

2.1 DDS Fundamentals

Publish/Subscribe model deals with publisher that sends messages to a particular topic from where a subscriber receives the message of its interest. Publishers and subscribers may not necessarily know each other in such model rather they can contact with a specific topic from where they can publish and subscribe for particular data. In this way publish/subscribe model is more appropriate for distributed and large scale real time systems than client server communication model. There are four main components of publish subscribe model [50].

2.1.1 Domain

Domain here is a virtual network from where publish/subscribe model components send and receive data of their interest. Publishers and subscribers can communicate within the same domain with each other. Domain is very useful for making privacy for same community by isolating it from others to ensure optimized and safe communication of interests among its components.

2.1.2 Publisher

Publisher sends actual data to the topic through an object known as data writer. There can be multiple data writers of same publisher. Thus publisher can send data to multiple topics of various data types through different data writers.

2.1.3 Subscriber

Subscriber receives actual data from the topic through an object known as data reader. There can be several data readers associated with the same subscriber. Thus a same subscriber can receive data from different topics of different data types through different data readers. Data sent by publisher to an application is first processed by subscriber than data sample is stored in data reader.

2.1.4 Topic

Publisher and subscriber communicate with topic to send and receive data of their interest. Topic has a single name in terms of ID or key and a data type. Topic has a specific QoS parameter based on which data readers and data writers interact with it for their required data. For instance, a topic named as "pressure" can be used to store several samples of pressure that is taken by distributed pressure sensors. Now the sensors that have taken the measurements can be publishers and sensors that make further calculations based on these measurements may be subscribers of the samples.

2.2 DDS Advantages

DDS is a network based middleware that yields the transmission of messages, commands and events over the network. The main features of DDS middleware [51] are presented below:

2.2.1 Data centricity

DDS is based on Global data Space for distributed applications which makes it data centric middleware. The advantage of being data centric is reduction of delay in communication that prevents failures.

2.2.2 Connectionless Servicing

Connection less service is embedded in DDS Real Time Publish Subscribe (RTPS) protocol so there is no need for (Point to Point) P2P connection within the network. This allows DDS to integrate large amount of devices and applications with decreased cost.

2.2.3 Auto Discovery

Nodes detect each other automatically within the network due to Data Centric Publish Subscribe Model (DCPS) in DDS which offers thorough decoupling of publishing and subscribing applications. DDS offers automatic discovery structure that allows domain participants to find topics and establish communication without the need to know about other participants.

2.2.4 Interoperability

Interoperability is the major quality of DDS based middleware. It was designed to provide interoperability especially in mission critical and real time constrained environment. Interoperability is delivered through DDS QoS, Application Programming Interface (APIs) and RTPS wire protocols. DDS APIs provides wide range of OSs and programming options for the developers. RTPS wire protocol defines platform independence and auto discovery whereas, QoS provide certain options of QoS policies to be adopted to specify communication behavior.

2.3 QoS policies

There are huge set of QoS policies [50] provided by Object Management Group (OMG) in DDS that controls and manages outcomes according to communication requirements. Through these policies one can control some of the features related to the behavior of DDS by setting transport layer requirements associated with reliability and efficiency to achieve desired behavior of network traffic. But to set quality of services for one operation is a challenge as some of them may contradict with each other. For example 'time based filter' and 'deadline' , both of them associated with time where 'deadline' depends upon specified time limit for data to be received by Data Reader and 'time based filter' deals with minimum separation time. These two may conflict with each other if not set carefully and in consequence the system will not work properly. The QoS used in experimentation are presented below.

2.3.1 DURABILITY

Data writing time may be selected through this QoS policy that whether data should outlive after it has written by publisher and collected by subscriber for late joining domain participants or not. By setting this QoS to TRANSIENT we can have data (even after it has been served to Data Readers) by maintaining a record into memory, thus data is not bound to Data Writer life cycle. Whereas if DURABILITY QoS policy is set to VOLATILE than data is not stored in memory on behalf of Data Writers thus it can save resources for new incoming data.

2.3.2 LATENCY_BUDGET

This QoS defines maximum acceptable delay among data sending and receiving. This policy provides us guidelines to service time limits and give us flexibility to get changed at any time. If a set delay cannot catch an update the service will not remove the packet or raise any flag.

2.3.3 DEADLINE

It defines a specific value of time after which a frame must be received by the subscriber. It can also be used to signify degradation of network performance. If a subscriber cannot get frames within specified deadline period that means system is malfunctioning and subscriber must get data from another topic of same characteristics.

2.3.4 RELIABILITY

It offers requested reliability level to subscriber and publisher. Levels of reliability can be set by two aspects, RELIABLE and BEST_EFFORT. RELIABLE level deals with the acknowledgement sent by Data Reader after successfully receiving every packet and Data Writer will keep a copy of sent data packet until an acknowledgment is received and if acknowledgement is not received by Data Writer it will send the copy of same data after certain defined time. But it has a negative impact on channel bandwidth for sending and receiving acknowledgements and also on latency because subscriber has to ensure order and integrity before sending acknowledgements. When packets are continously keep droping during transmission than its better to use BEST_EFFORT that does not need acknowledgments at all for example in video streaming.

2.3.5 TIME BASED FILTERING

It deals with the rate of data samples delivered to Data Reader within specified deadline period. Minimum separation time can be set in this QoS policy to filter and drop packets for Data Reader. Middleware service will drop any new data packets if the same instance of data is received within specified minimum separation time. Contradiction arrives when minimum separation time is greater than deadline period so data packet must be received within deadline period but after minimum separation time which is depicted in figure. In our experimental work TIME BASED FILTER is analysed to examine computational resource usage and bandwidth by only sending required data samples to Data Readers and filter packets that are received faster than prescribed rate.

2.3.6 HISTORY

It defines the actions of middleware service if the value of data modifies before reaching to reciever. The management of frames sent by Data Writer of publisher and received by Data Reader of subscriber can be controlled by system through this QoS policy. Data Writer may keep data to itself on behalf of Data Reader and on the other hand Data Reader may keep data to itself until it is read by subscriber. KEEP_LAST value of HISTORY QoS policy can be set at Data Writer's end of publisher with one picture and with group of pictures (GOP) at Data Reader's end of subscriber because some times subscriber's decoder need some previos data frame values to decode.

2.3.7 RESOURCE_LIMIT

It deals with the measurment of consumption of resources by DDS in order to adhere with QoS requirements. RESOURCE_LIMIT must be set considering other QoS parameters values otherwise it will affect their performance too. For instance, if RELIABILITY QoS is set to RELIABLE then Data Writer must some space to store data samples to receive their acknowledgments. For example 'max_samples_per_instance' is 1 then there will be no space for Data Writer to store data samples, in this RELIABILITY QoS will suffer by RESOURCE_LIMIT's wrong settings.

2.3.8 LIVELINESS

It deals with the on and off state of any particular entity within DDS. Every entity is continously signaled to know its existence and non existence time. Some times there are sleeping algorithms embedded to obtain efficient energy utilization. LIVILINESS is very important for publisher and subscriber to know about their active time in order to perform reliable and efficient communication of fresh data. The period of signalling should not exceed 'liveline_lease_duration' other wise Data Reader may misinterpret Data Writer's active schedule time.

2.4 Mapping of QoS to communication requirements of Smart grid

As we have discussed it earlier that DDS middleware is the best for real time mission critical distributed applications that also provides interoperability among heterogeneous devices and applications. It also covers vast range of Samrt grid communication requirements that are explained above with the help of rich set QoS policies. Table 2.1 to Table 2.3 shows the mapping of DDS QoSs with communication requirements with reasoning.

Table 2.1: Mapping QoS policies over Smart grid communication requirements

Communication Requirements	QoS Policies	Reasoning
Distributed communication	USER_DATA, TOPIC_DATA, GROUP_DATA	DDS is independent of network and routing protocol. It may face dynamic protocols over the network and these QoS policies deal with unknown data and routing.
	TRANSPORT_PRORITY	TRANSPORT_PRIORITY is useful for this communication requirement because it helps Middleware choosing the type of protocol to send data.
Latency	DEADLINE	DEADLINE QoS policy helps to allocate time period for at least one sample of data to be transmitted to control the latency of data on network.
	LATENCY_BUDGET	LATENCY_BUDGET is used to control the maximum acceptable delay or latency.
	TIME_BASED_FILTER	Minimum separation time between samples of data is calculated and controlled by TIME_BASED_FILTER QoS policy so it controls speed of data and offers specified amount of latency.
	CONTENT_BASED_FILTER	CONTENT_BASED_FILTER QoS policy does not allow unwanted data, it saves time by processing only desired data, thus minimize latency.
Reliability	PRESENTATION	PRESENTATION QoS policy deals with the changes being occur in data instances and it also keeps track of order of changes so that the data we receive is fresh and reliable.
	DEADLINE	DEADLINE ensures the new value of data each within deadline period so data we receive is fresh and we can have surety of data reliability.
	RELIABILITY	RELIABLE value from RELIABILITY QoS makes sure of successful data transfer no matter if the data is lost during transmission so it keeps the copy of data for future to ensures data transmission reliability.
	LIFESPAN	LIFESPAN QoS specifies the validity of data so we can overcome unreliable data through this Qos policy.
Data rate	DEADLINE	DEADLINE ensures the new value of data each within deadline period so data we receive is fresh and we can have surety of data reliability.
	TIME_BASED_FILTER	TIME_BASED_FILTER QoS policy determines minimal time period among two data samples. We can change this separation period according to demand of application thus may control the data rate of application.
	CONTENT_BASED_FILTER	Redundant data affects our high data rate severely so CONTENT_BASED_FILTER QoS policy is useful in eliminating redundant data and allow us to select data of our choice to minimize the affects on data rate.
Bandwidth	CONTENT_BASED_FILTER	CONTENT_BASED_FILTER also very helpful in utilizing bandwidth efficiently. Redundant data affects our high bandwidth severely, this QoS policy eliminates redundant data and allow us to use bandwidth effectively.

Table 2.2: Mapping QoS policies over Smart grid communication requirements

Communication Requirements	**QoS Policies**	**Reasoning**
Throughput	DEADLINE	DEADLINE ensures the new value of data each within deadline period and we can define this deadline period of our choice, we can have the desired throughput by setting deadline period.
	TIME_BASED_FILTER	TIME_BASED_FILTER QoS policy determines minimal time period among two data samples. We can change this separation period according to the demand of application thus may control the throughput of network efficiently.
	CONTENT_BASED_FILTER	CONTENT_BASED_FILTER QoS policy does not allow unwanted data we can achieve efficient throughput by only receiving desired data.
	RELIABILITY	RELIABLE value from RELIABILITY QoS makes sure of successful data,transfer no matter if the data is lost during transmission so it keeps the copy of data for future to ensure high and efficient throughput.
Interoperability	USER_DATA TOPIC_DATA GROUP_DATA	Middleware is not concerned with the type of data so USER_DATA, TOPIC_DATA and GROUP_DATA QoS policies are helpful in achieving the interoperability.
Flexibility	TRANSPORT_PRORITY	TRANSPORT_PRIOROTY fulfills the requirement of flexibility of choosing best network with good band-width for data transmission.
	USER_DATA TOPIC_DATA GROUP_DATA	Middleware is not concerned with the type of data so USER_DATA, TOPIC_DATA and GROUP_DATA QoS policies are helpful in achieving the flexibility.
Scalability	USER_DATA	DDS does not concerned with type and size of data travels over the network from any devices, thus USER_DATA QoS policy ensures scalability.
	RESOURCE_LIMITS	Scalability can also be controlled through RESOURCE_LIMITS QoS policybecause it controls the resources given to application, it delivers scalability through this useful QoS policy.
Security	LIFESPAN	LIFSPAN ensures the life of data that we can alter by our choice as it is one of the procedure to ensure security. If data is old from a specified time period than it is not considered as secured.
Complexity	DESTINATION_ORDER	The logical order of changes occur in Middleware is determined by DESTINATION_ORDER QoS policy so this QoS policy eases our life by sorting out the changes and making them in order to be processed easily.
Robustness and availability	DURABILITY	DURABILITY QoS policy determines whether the data should be outlived its writing time or not so it offers data availability even after its life span.
	LIVELINESS	LIVELINESS QoS policy tells us about the entity whether it is active or not thus informs us about its availability.
	ENTITY_FACTORY	It deals with the side effects of certain operations and let us know about the system failures, thus helps in making the system robust.

Table 2.3: Mapping QoS policies over Smart grid communication requirements

Communication Requirements	**QoS Policies**	**Reasoning**
Efficiency	DEADLINE	DEADLINE QoS policy controls the latency of data on network thus enhancing efficiency of overall system.
	LATENCY_BUDGET	LATENCY_BUDGET is used to control delay to ensure maximum efficiency of the system.
	OWNERSHIP	OWNERSHIP controls the ownership of DDS entities to write the data over instances. Depends on system behavior the ownership can be changed time to time so OWNERSHIP QoS controls and orderly behavior of transfer of ownership to various entities thus provide maximum efficiency to the system.
	CONTENT_BASED_FILTER	CONTENT_BASED_FILTER as described above eliminates unwanted data that affects our high data rate and system performance severely so this QoS policy is useful for ensuring high efficiency.
Predictability and intelligence	LIVELINESS	The information about the status of an entity is delivered by LIVELINESS QoS policy that whether it is active or not so that system can make timely predictions and intelligent decisions based on report of an entity.
	HISTORY	The HISTORY QoS policy specifies the behavior of Middleware and intelligently determines data sample value to deliver based on arrival time.
	ENTITY_FACTORY	ENTITY_FACTORY controls the operation of various entities thus provide certain level of intelligence and predictability to system.

CHAPTER 3

LITERATURE REVIEW

This chapter highlights some of the works involving real time distributed environment and discusses the suitability of DDS standard and wireless channels in industrial automation scenarios.

An, Kyoungho, et al. [52] studied the performance of QoS providing publish/subscribe based middleware to monitor resources in cloud. They have proposed fast and efficient middleware approach in place of web based systems. They pointed out that web based platforms such as SOAP and Restful APIs cannot handle such large amount of data for real time monitoring of applications. Moreover, web based platforms incur such overheads that degrades efficiency and reliability that may affect over all network performance. Authors suggested a publish/subscribe middleware to overcome these challenges for efficient resource monitoring and data handling in cloud platforms. In the end authors did several performance tests and compare middleware with web based platforms and claimed that middleware approach is far more better than conventional systems.

Xiong et. al [53] described three possible architectures based on DDS and evaluated the implementation of those platforms to investigate their design trade-offs. They compared the performance of their implementations with each other and with other Publish/Subscribe middleware. It was concluded, based on the experimentation that DDS based implementations perform substantially better than their non-DDS counterparts and are generally well suited for real-time data-critical distributed environment.

In [54] Poza et. al realized that QoS support usually available in communication layer merely delivers simple networking parameters like message delay or congestion control and, therefore, are insufficient in such scenarios where information optimization, real-time support or component abstraction is desired. They proposed a middleware architecture, named Frame Sensor Adapter to Control (FSA-Ctrl), consisting of two layers: a DDS based communication layer and the other Sensor Web Enablement (SWE) based control layer. Both layers sandwich a rich set of QoS policies that empowers control layer to take important decisions about distributed questions like component mobility or information redundancy detection. They implemented their proposed architecture on a home automation problem. Although DDS is a very powerful and flexible technology but it may also prove to be rather complex to fully comprehend, particularly for end users. This issue was spotted and dealt with by Calvo et. al in [55]. The authors floated the idea of a software component encapsulating the functionality of commonly used industrial automation controllers like PLCs, IPCs and Robots which can then be used to create any automation application. The role of DDS in this case can simply be as a communication backbone. The paper shows how to map different traffic patterns using DDS entities taking full advantage of DDS QoS policies.

Abdel Rahman et al. [56] surveyed about Message Oriented Middlewares (MOM) for real time and distributed environment in smart grid. So far this is the only paper that analyzes the smart grid middlewares based on their functionality and performance. They illustrate MOM in publish subscribe paradigm (PSMOM) that delivers high scalability and asynchronous nature for many to many communication infrastructure. They have also considered the support of QoSs in MOM to analyze its role in system performance. They have studied and analyzed three of major industrialized QoS providing middlewares for smart grid.

- Extensible Messaging and Presence Protocol (XMPP) is related to XML and provides request response services, presence and real time messaging. It supports QoS functionalities and is a best effort protocol augmented with some extension protocols (XEP). The extended protocol (XEP-0203) provides timestamp information related to stored messages that can be useful in case of slow delivery. The rules to handle time sensitive messages of an application can be defined with the help of advanced message processing extensions (XEP-0097). Negotiating XML streams compressions are delivered through (XEP-0138) and the priorities for resources that are connected are specified by (XEP-0168) protocol. There are several other protocols that it supports.

- RabbitMQ is based on Advanced Messaging Queue Protocol (AMQP) and is an open source message broker. It addresses wire protocol and the protocol that defines AMQP implementation semantics. In this way these implementations become interoperable with other implementations. Brokering task defined by AMQP is divided among message queues and exchanges which is identical to router that is based on a set of rules (deciding message routing queues) and accepts incoming messages.

The third middleware specifies in this paper is DDS which is concluded as a most suitable technology for SG applications that is already explained.

Kai Shi at el. [57] implemented data communication platform in smart micro grid. It is implemented as a wind turbine real-time monitoring system to show the feasibilities and advantages of using DDS middleware into smart micro-grid monitoring over the IP Network using TCP/IP and UDP protocols. A web server is built to provide the interface for user to monitor and analyze data. Real time data base monitoring for forecasting, controlling and historical data base is used to monitor data for play back analysis, system performance and energy efficiency. MySQL DBMS is used to store data collected from generation and consumer side along with RTI Connect to provide publish subscribe paradigm. DSpace board is connected to wind turbine to collect and monitor data thus provide data acquisition system. The data publisher can detect new coming data and send them out into the DDS domain in the network. RTI Real-Time Connect (RTC) service works at the background together with the realtime database. This service worked seamless with the MySQL database server and is able to subscribe the data of interest from DDS domain and simultaneously store the data into the real-time database. The real-time data monitoring service performed in web server. This web application read the latest data from the real-time database every second and graphically displayed these data to users for monitoring and analysis purpose. Users can access the web site when their devices are connected into the network.

L. Jiang et al. [58] described and tested the deployment feasibility of substation automation standard IEC 61850. They have designed a data model to reduce complexities and difficulties of IEC 61850 implementation by considering Abstract Communication Interface (ACSI) for interoperability. The GOOSE messages from different electric devices are taken from the standard to perform experimental work and establishing communication among these devices and some performance tests are conducted. REST (Representational State Transfer) Services are used alongwith IEC 61850 data model to demonstrate delivery of abstraction and interoperability enhancement. They have proposed an approach to enable WAN in control systems and electric power telemonitoring IEDs for high speed transmission and reliable exchange of data.

CHAPTER 4

DDS BASED IMPLEMENTATION OF SMART GRID DEVICES USING ANSI C12.19 STANDARD

4.1 ANSI C12.19

ANSI C12.19 is a standard developed by American National Standards Institute (ANSI), Inc. C12, WG2 and SC17 [56]. The standard is developed for Smart grid utility industry to provide standardized end device data tables. This standard is sponsored, checked and approved by National Electrical Manufacturers Association (NEMA) through standards development process. ANSI C12.19 delivers a common data structures to transfer data among Smart grid utility end devices specially in meters and other user/customer devices. Standard data structures are represented in terms of table sets that are joined together to form sections known as decades. Every decade is related to a particular feature or function for example load profile or usage time etc. Table data is transmitted to or from one end device to another end device by writing or reading to a specific table either as a full or a portion. Standardized tables provided by this Standard can be utilized for reading and programming purposes of an end device. These tables supply data control parameters that can be implemented in large amount of applications and programs. For our implementation and experimentation purposes, we have selected two data structures from this Standard. One is 'Electric Element Description' which is related to consumption side such as Smart Meter and other is 'Utility Information' which is related to transmission or distribution side. Through these two topics, we can measure the performance in transmission and consumption sides of Smart grid. The definition of these structures are shown in Figures 4.1 and 4.2.

```
struct Electric_Element_Descriptions
    {
        uint    E_FREQ;
        uint    E_NO_OF_ELEMENTS;
        uint    E_BASE_TYPE;
        uint    E_ACCURACY_CLASS;
        uint    E_ELEMENTS_VOLTS;
        uint    E_ED_SUPPLY_VOLTS;
        string  E_CLASS_MAX_AMPS;
        string  E_TA;
        string  E_KH;
        string  E_KT;
        uint    E_INPUT_SCALAR;
        string  E_ED_CONFIG;
        uint    E_ELEMENTS;
        uint    E_VOLTS;
        uint    E_AMPS;
    }
```

Figure 4.1: Electric Element IDL Definition

```
struct Utility_Information
{
    string     OWNER_NAME;
    string     UTILITY_DIV;
    string     SERVICE_POINT_ID;
    string     ELEC_ADDR;
    string     DEVICE_ID;
    string     UTIL_SER_NO;
    string     CUSTOMER_ID;
    string     TARIFF_ID;
    string     EX1_SW_VENDOR;
    uint       EX1_SW_VERSION_NUMBER;
    uint       EX1_SW_REVISION_NUMBER;
    string     EX2_SW_VENDOR;
    uint       EX2_SW_VERSION_NUMBER;
    uint       EX2_SW_REVISION_NUMBER;
    string     PROGRAMMER_NAME;
    string     MISC_ID;
    }
```

Figure 4.2: Utility Information IDL Definition

4.2 ANSI C12.19 implementation over DDS

Interface of software component is defined by a specification language known as Interface Description Language (IDL). It provides a neutral way to describe language interface for communication of two different software components that have separate language support for example components written in C++ and JAVA. IDL is based on Remote Procedure Call (RPC) software that is very useful to link machines with different OS, languages and architecture. For our experimental work setup we have constructed IDL files using two topics that are shown in Figures 4.1 and 4.2 respectively. The details of these topics are presented in Tables 4.1 and 4.2.

We have develop a methodology so far based on our IDL file to establish communication among publish subscribe Smart grid applications or devices.

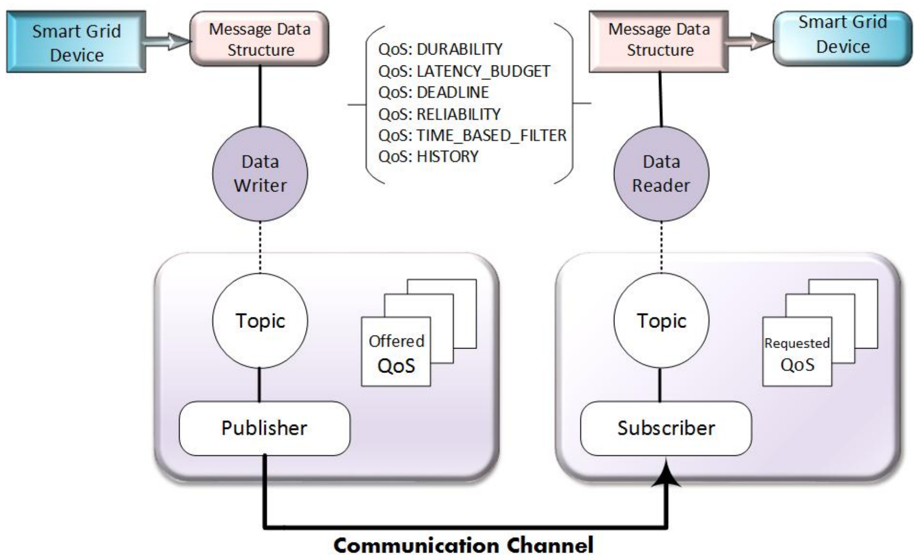

Figure 4.3: Standard implemetation over RTPS DDS model

Figure 4.3 depicts over all behavior of DDS in which Smart grid device will construct its data structures based on ANSI C12.19 standard. When there is a need to transmit some data, one of the Smart grid device will deliver these data structures to Data Writer. The Data Writer analyzes the data whether it is according to structure of specified topic, if so then Data Writer writes the data on related topic. Data writer will not write the data unless the topic corresponds to its specified structure. Now its time for publisher to publish the data over a communication channel. Publisher will publish data based on QoSs it offered. The system performance might be different based on certain QoS parameters settings. Similarly on the other side there will be a subscriber waiting for the data that also has certain requested set of QoS parameters. QoS parameters settings on both ends is very subtle and difficult because some times QoS settings may contradict with each other that may result in incorrect outcomes that are very crucial in real time distributed applications. In this case both the devices deals with Smart grid real time distributed system so one must be careful for QoS settings at each deployment. Subscriber will subscribe data from publisher which is inserted in topic. Once the data is inserted in the topic, other Smart grid device that requires this data will read the data through Data Reader. The data structures delivered and obtained will be same instead of just one or two fields for example utility name, device name, time of delivery and time of retrieve. The Smart grid device will now get data for further processing. In the same way any Smart grid device can become publisher and subscriber at a time because whole Smart grid communication infrastructure is based on full duplex mode.

4.2.1 Performance Metrics

Performance measurement is necessary for adapting state of the art DDS publish subscribe middleware technology for reliable data communication among Smart

Table 4.1: Electric Element detail for topic 1

Variable Name	Standard Data type	DDS IDL primitive data type	Value	Description
E_FREQ	UINT	uint	0-7	Power frequency rating code
E_NO_OF_ELEMENTS	UINT	uint	0-7	Number of commodity measuring elements per measuring input.
E_BASE_TYPE	UINT	uint	0-15	Mentions meter base type
E_ACCURACY_CLASS	UINT	uint	0-63	Solid state meter standard accuracy class definition.
E_ELEMENTS_VOLTS	UINT	uint	0-15	Meter element voltage code to define voltage class.
E_ED_SUPPLY_VOLTS	UINT	uint	0-15	External supply voltage code to define end device supply voltage.
E_CLASS_MAX_AMPS	STRING	string	-	IEC maximum ampere ratings.
E_TA	STRING	string	-	RMS amperage test amperes (TA) for testing and adjusting meter
E_KH	STRING	string	-	Represents Watt hours per revolution.
E_KT	STRING	string	-	Commodity amount selected for test pulse output.
E_INPUT_SCALAR	UINT	uint	-	Divisor for input values
E_ED_CONFIG	STRING	string	-	Form number per ANSI C12.10 characters
E_ELEMENTS	UINT8	uint	-	Use to define E Elements bit field.
E_VOLTS	UINT	uint	-	Use to define E Volts bit field.
E_AMPS	UINT	uint	-	Use to define E Amps bit field.

grid devices.

1. **Latency:** It is the addition of, time taken by the data packet to reach at receiver's side and time taken by data acknowledgement to reach at sender's side successfully. Latency is basically a sum of propagation and queuing delay at sender's side and some times equal to Round Trip Time RTT as given below.

$$Latency \approx RTT \tag{4.1}$$

Real time systems generally deal with three kinds, Firm real time, soft real time and hard real time. Firm real time and soft real time systems allow us some kind of flexibility as compared to hard real time in terms of

Table 4.2: Utility Element detail for topic 2

Variable Name	**Standard Data type**	**DDS IDL primitive data type**	**Value**	**Description**
OWNER_NAME	STRING	string	-	Upto 20 valid characters.
UTILITY_DIV	BCD	string	-	Utility division number up to 16 valid characters.
SERVICE_POINT ID	BCD	string	-	Up to 20 characters and ID number is attached to service point.
ELEC_ADDR	BCD	string	-	End device logical and electrical address for mapping and study purposes.
DEVICE_ID	BCD	string	-	Hardware ID up to 20 characters.
UTIL_SER_NO	BCD	string	-	Utility specified serial number up to 20 characters.
CUSTOMER_ID	BCD	string	-	All valid characters up to 20 max.
TARIFF_ID	STRING	string	-	Billing tariff identification
EX1_SW_VENDOR	STRING	string	-	Manufacturer/software developer's name.
EX1_SW_VERSION NUMBER	UINT8	uint	0-255	Software programming or configuration version number.
EX1_SW_REVISION_NUMBER	UINT8	uint	0-255	Software programming or configuration version number.
EX2_SW_VENDOR	STRING	string	-	Manufacturer/software developer's name.
EX2_SW_VERSION_NUMBER	UINT8	uint	0-255	Software programming or configuration version number.
EX2_SW_REVISION_NUMBER	UINT8	uint	0-255	Software programming or configuration version number.
PROGRAMMER_NAME	STRING	string	-	Name of last programmer or programming device.
MISC_ID	STRING	string	-	Up to 30 valid characters used for verification number, approval number and bar code.

events occuring. Although deadline period is defined for packet receiving time but system will not fail if there is any longer delays or packet loss ocassionally [57]. Our experimentation deals with time stamps of every packets based on clock at publisher side. The publisher after successfully receiving acknowledgement calculates RTT as difference of time between packet sending and acknowledgment receiving. To escape from any uncertainty, we have used built-in RTI connext modules to calculate latency.

2. **Jitter:**Variation in latency is known as jitter. If there are same amount of delays then it means jitter values are very less during the flow of packet from sender to receiver. Mathematical formulation of jitter is:

$$Jitter = \sqrt{\frac{1}{n}\sum_{a=1}^{n}(a_i - \overline{a})} \tag{4.2}$$

   Where 'n' is total number of delayed samples and $\overline{a}$ is the mean value of 'n'. Precision of system can be calculated easily through jitter. During transmission if there is small amount of latency but large amount of jitter in channel than it depicts that there are some packets taking unusual longer time to be received by subscriber. Than reliable flow of updates are difficult to send within that duration of latency.

3. **Throughput:**Average rate of data transmission over a channel is called throughput. This data not also considers protocol overhead alongwith paylod so that effective channel utilization can be observed through throughput.

$$Throughput = \frac{No.of Packets * PacketSize}{TotalTime} \tag{4.3}$$

   If system is giving high throughput it means bandwidth is being utilized efficiently over the channel. But throughput beyond certain threshold level may result in packet loss and subsequent congestions which cause longer delays. That is why throughput monitoring is much essential in real time mission critical applications.

4.3 Smart grid entities communication scenarios

Based on DDS components and architecture, we have made different scenarios for implementation of Smart grid devices. In these scenarios, there is a Smart grid control station on one side and on the other side there are end devices, which will keep increasing. Same communication channel is used for all scenarios.

Our first two scenarios depicts the communication of Electrical Utility divisions from two different companies with Smart grid control and monitoring station. These Electrical Utility divisions send data control parameters to control and monitoring station based on Utility information IDL definition of topic which is depicted in Figure 4.2. The communication among these two entities is carried out through DDS publish subscribe platform where publishers from Utility divisions write data through Data Writer on communication channel and control and monitoring station read the data through Data Readers of subscribers. Scenarios 1 and 2 are shown in Figures 4.4 and 4.5.

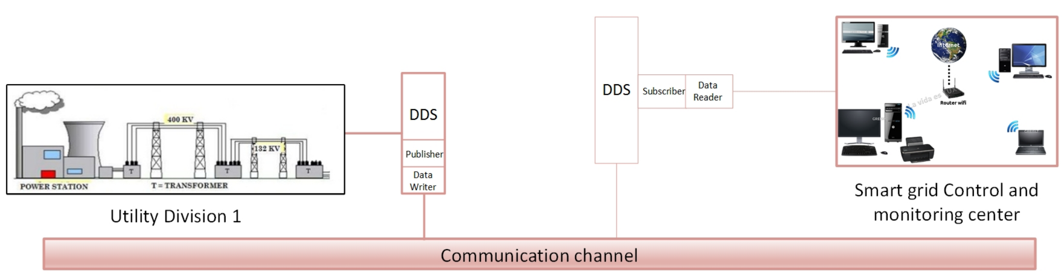

Figure 4.4: ANSI implementation Scenario 1

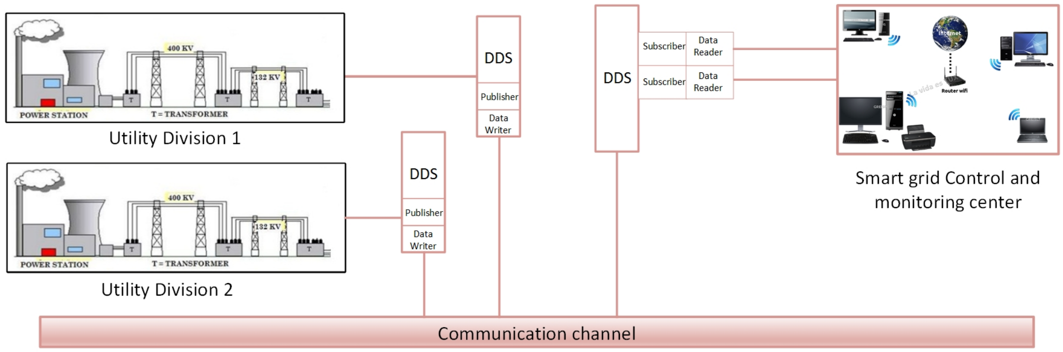

Figure 4.5: ANSI implementation Scenario 2

In scenario 1, there is only one Publisher and 1 Subscriber whereas in scenario 2, Publishers and Subscribers are increased to 2 with increase in utility divisions. Similarly in other four scenarios we have added Electrical element devices along with Utility divisions to analyze the communication behavior on the same communication channel. Electric elements sends the data control parameters based on Electric element IDL definition of topic depicted in Figure 4.1 to control and monitoring station. Scenario three is depicted in Figure 4.6 while other scenarios are depicted in Figures 4.7, 4.8 and 4.9 where electrical devices as meters from different companies are added one by one. Number of Publishers and Subscribers have also increased throughout the scenarios along with entities. As per scenarios, all of these entities belong to different companies but follow the same standardized data sets, so each of the entity has its own publisher to transmit data and there are separate subscribers at control and monitoring station to subscribe data from these various entities. In DDS domain we can have as much subscribers as we need if their topic structure is same. For our study we have considered number of subscribers equal to number of publishers to analyze performance easily.

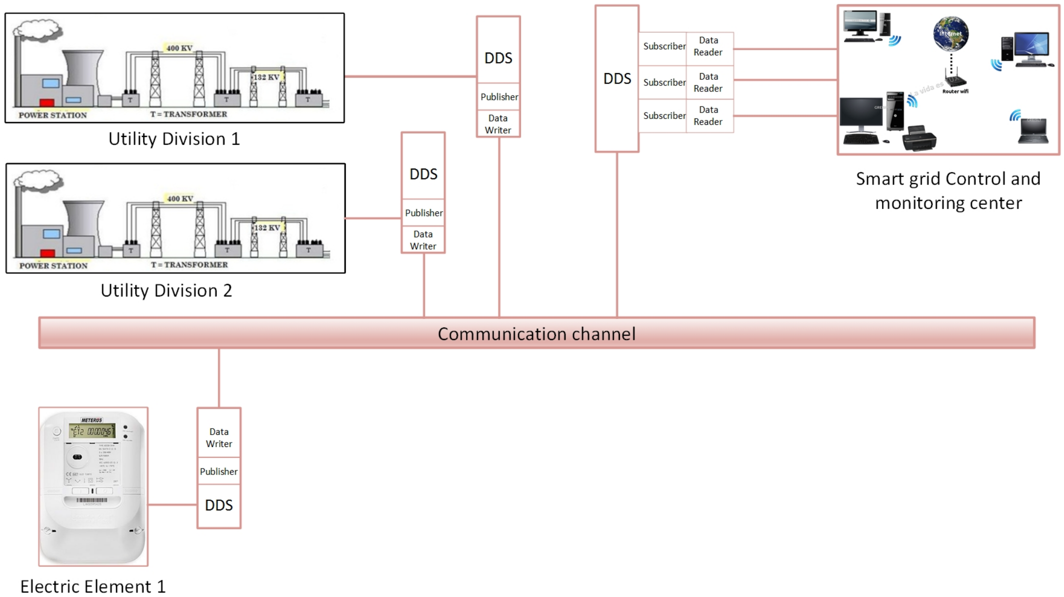

Figure 4.6: ANSI implementation Scenario 3

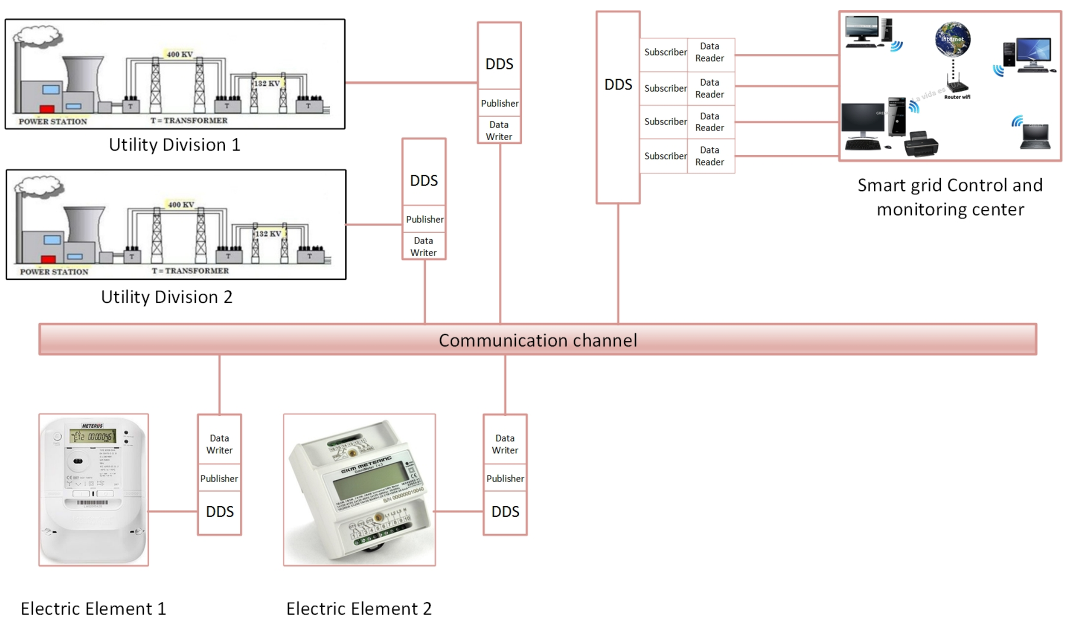

Figure 4.7: ANSI implementation Scenario 4

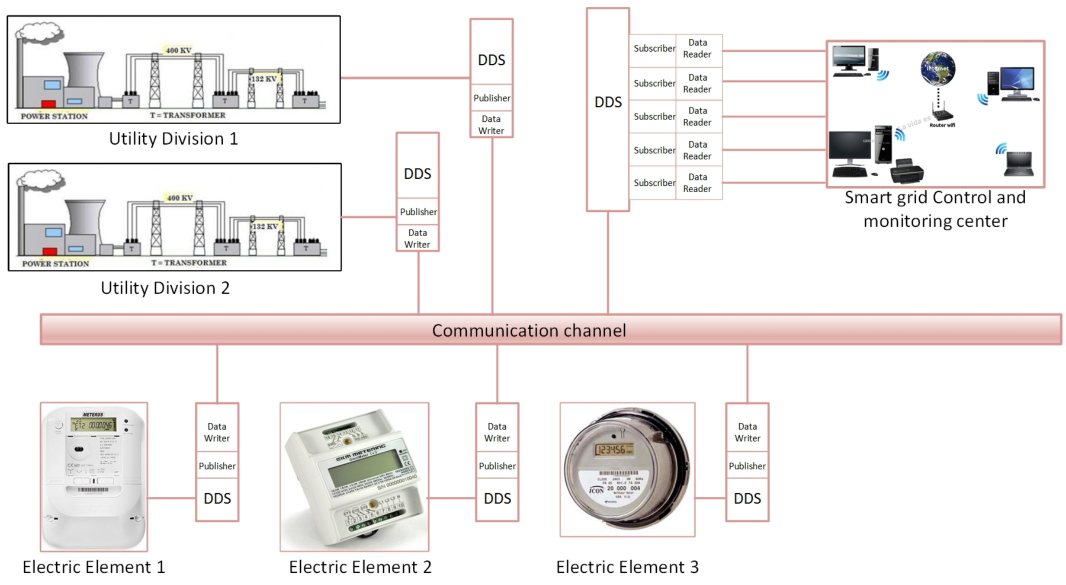

Figure 4.8: ANSI implementation Scenario 5

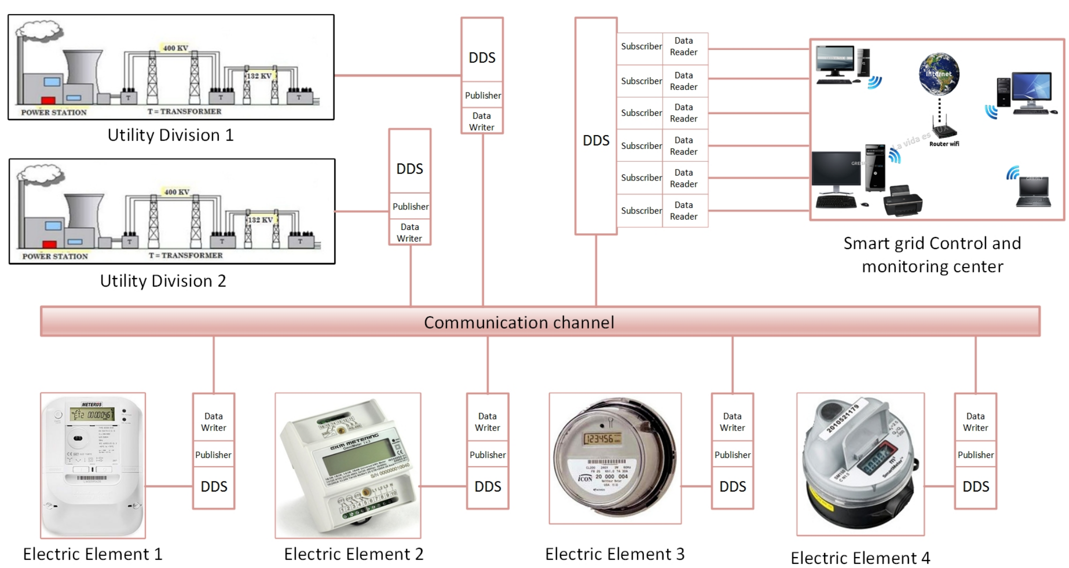

Figure 4.9: ANSI implementation Scenario 6

Throughout the experiments our number of publishers and subscribers are being increased. This behavior is necessary to measure the accurate performance of DDS as it is also in favor of predictability and scalability.

4.4 Experimental Setup

The hardware and software specifications used in experimental work is shown in Table 4.3. The experiment is performed in one of the labs in KFUPM Computer Engineering department.

Table 4.3: Hardware and Software specifications

Hosts/ Specifications	**Machine 1 (Publisher)**	**Machine 2 (Subscriber I)**	**Machine 3 (Subscriber II)**
CPU	Intel(R) Core 2 Duo CPU P8800 @ 2.66 GHz	Intel(R) Core 2 Duo CPU P8800 @ 2.66 GHz	Intel(R) Core 2 Duo CPU P8800 @ 2.66 GHz
OS	Window 7 64 bit	Window 7 64 bit	Window 7 64 bit
Memory	16 GB	16 GB	16 GB
Network	LAN/Wifi 100 Mbps	LAN/Wifi 100 Mbps	LAN/Wifi 100 Mbps

Table 4.4: Default QoS policies of publishers and subscribers

QoS Policy	**Publisher Value**	**Subscriber Value**
DURABILITY	VOLATILE	VOLATILE
RELIABILITY	RELIABLE	BEST_EFFORT
HISTORY	KEEP_LAST	KEEP_LAST
RESOURCE_LIMITS	LENGTH_UNLIMITED	LENGTH_UNLIMITED

Table 4.5: Modified QoS policies of publishers and subscribers

QoS Policy	**Publisher Value**	**Subscriber Value**
DURABILITY	TRANSIENT_LOCAL	TRANSIENT_LOCAL
RELIABILITY	BEST_EFFORT	RELIABLE
HISTORY	KEEP_ALL	KEEP_LAST
RESOURCE_LIMITS	LENGTH_UNLIMITED	1

We have used certain QoS policies in our experimentation work through which the behavior of system is analyzed with the help of default and modified QoS policy values. The default values for these QoS policies can be seen in Table 4.4, while the modified QoS policies values are shown in Table 4.5. By changing the values of QoS policies we have analyzed that we can change the behavior of our system according to our own requirements. Default value for DURABILITY QoS policy is VOLATILE for both publishers and subscribers, that means middleware does not keep any data samples for late joining participants due to which scalability can be affected. This value is modified to TRANSIENT_LOCAL, because it allows us to deliver data samples to any late joining new participant. In this way any late joining participant may become part of communication system if it holds same features of declared topics. Default value of RELIABILITY QoS policy is RELIABLE for publishers and BEST_EFFORT for subscribers. These values are swapped in publishers and subscribers to analyze latency and throughput behavior at both sides. In case of BEST_EFFORT the Data Writer is free to write all the time without noticing the queue size of Data Reader thus it increases latency and throughput, while RELIABLE ensures the successful data transfer to subscriber and limits Data Writer's queue to 50 samples only that results in low latency. HISTORY and RESOURCE_LIMITS QoS policies are chosen to support RELIABILITY QoS because for RELIABLE value, it has to resend the data samples according to HISTORY and RESOURCE_LIMITS QoS settings.

Simulation behavior according to Figure 4.3 is implemented by employing one to one and many to many communication model. All the domain participants are considered to be in single domain. We have used three machines at a time in which one is used as a publisher device of Smart grid and other two machines as subscriber I and subscriber II respectively. First we perform communication test by sending and receiving different packets of strings on these machines. Latest version of RTI DDS Connext 5.2.3 is used to establish communication for both publisher and subscriber that has two topics shown in Figures 4.1 and 4.2. Ostinato is an open source tool which is used to generate network traffic at the rate of 10 Mbps to load the network to get the realistic performance results. C++ code is generated for publishers and subscribers using rtiddsgen utility on Visual Studio 2012 while Wireshark 1.2.3 and RTI performance test tools are used to do performance measurement over LAN and WiFi.

4.5 Results and Analysis

Random data values are generated at publisher's side and sent through a communication channel to one or more subscribers. The simulation is run on machines specified in Table 4.3 that are connected through LAN and WiFi. The generated values are transmitted among these machines and required numeric values are taken and stored for analysis.

Table 4.6: Latency performance with default QoS settings

Scenarios	**LAN**				**WiFi**			
P-S	**Latency Min (ms)**	**Latency Max (ms)**	**Latency Mean (ms)**	**Std. Dev.**	**Latency Min (ms)**	**Latency Max (ms)**	**Latency Mean (ms)**	**Std. Dev.**
1-1	7.8271	12.7639	9.2689	43.52	7.2536	13.2871	9.1283	52.15
2-2	7.9917	14.2877	9.9927	41.29	7.7712	14.1132	10.6148	55.27
3-3	8.7113	15.8376	11.2894	54.21	8.0182	15.2665	12.0011	41.20
4-4	9.0285	16.3679	13.8932	62.38	9.9175	17.2651	13.7651	47.89
5-5	10.2643	18.1661	14.9028	56.81	10.5254	17.8917	14.8716	42.01
6-6	11.4325	20.3718	15.8929	45.29	11.7355	19.9013	15.5562	33.28

Table 4.7: Latency performance with modified QoS settings

Scenarios	LAN				WiFi			
P-S	**Latency Min (ms)**	**Latency Max (ms)**	**Latency Average (ms)**	**Std. Dev.**	**Latency Min (ms)**	**Latency Max (ms)**	**Latency Average (ms)**	**Std. Dev.**
1-1	9.3672	15.6327	12.7253	65.16	9.8981	15.7253	12.5137	60.65
2-2	9.9816	17.2737	14.6927	31.27	10.3673	17.9873	13.9001	49.18
3-3	10.7365	18.3864	15.9185	53.84	12.3379	19.3720	16.1526	48.29
4-4	11.2657	20.8509	17.8135	56.28	13.8276	21.0019	17.7256	39.97
5-5	13.1736	22.1899	18.4162	46.53	15.9871	23.1739	19.2567	51.48
6-6	15.2637	24.7251	20.1427	50.17	16.9908	25.0190	20.8716	57.34

Table 4.8: Jitter performance with default and modified QoS settings

Scenarios	With Default QoS policies		With Modified QoS policies	
P-S	**LAN Jitter (ms)**	**WiFi Jitter (ms)**	**LAN Jitter (ms)**	**WiFi Jitter (ms)**
1-1	1.87	1.56	3.38	3.36
2-2	2.16	2.34	3.48	4.16
3-3	2.99	3.33	4.28	4.82
4-4	3.76	3.60	4.91	5.21
5-5	3.89	4.24	5.28	5.76
6-6	4.21	4.70	5.87	5.79

4.5.1 Latency and Jitter analysis

We used packet sizes from 100 to 200 Bytes based on data structure size in topics for our scenarios to perform latency and jitter tests. One to one and many to many model based experiments are performed and numeric values are taken to calculate latency and jitter. 12,000 to 60,000 packets are sent and received among machines and tests were repeated 10 to 15 times to get more precise results. Latency and jitter results are calculated according to the scenarios of publishers and subscribers shown in Figures 4.3 to 4.9. Tables 4.6 and 4.7 show the behavior of latency with default and modified QoS settings. The latency is calculated using equation 4.1. Similarly Table 4.8 shows the results of jitter that is calculated using equation 4.2. Figures 4.10 and 4.11 show the latency graph that contains comparison between default and modified QoS settings over LAN and WiFi. While Figures 4.12 and 4.13 show jitter graphs that contain the comparison between default and modified QoS settings over LAN and WiFi. Latency and jitter is calculated at publisher side while considering the additional network traffic load. It can be analyzed that latency and jitter obtained with default QoS policies values are lesser compared to modified QoS policies values. This behavior is obtained due to the RELIABLE value of RELIABILITY QoS policy at publisher side which is the default value, while modified value of RELIABILITY QoS policy that is BEST_EFFORT, offers more latency and jitter. The graph for each scenario can be examined where latency and jitter slightly increase when number of publishers and subscribers increase with both default and modified QoS policies settings. This is because of the increase in domain participants or traffic in single communication channel in a given point of time.

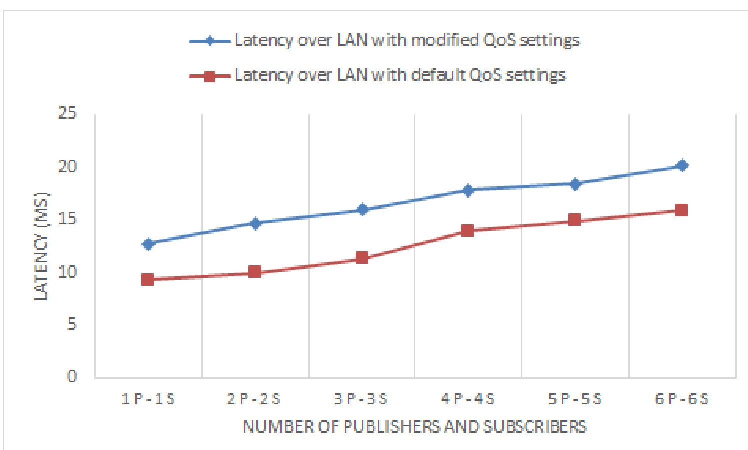

Figure 4.10: Latency comparison over LAN with default and modified QoS settings

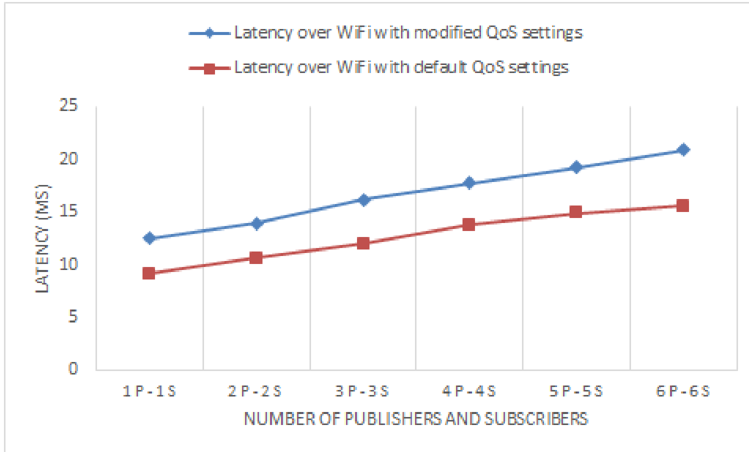

Figure 4.11: Latency comparison over WiFi with default and modified QoS settings

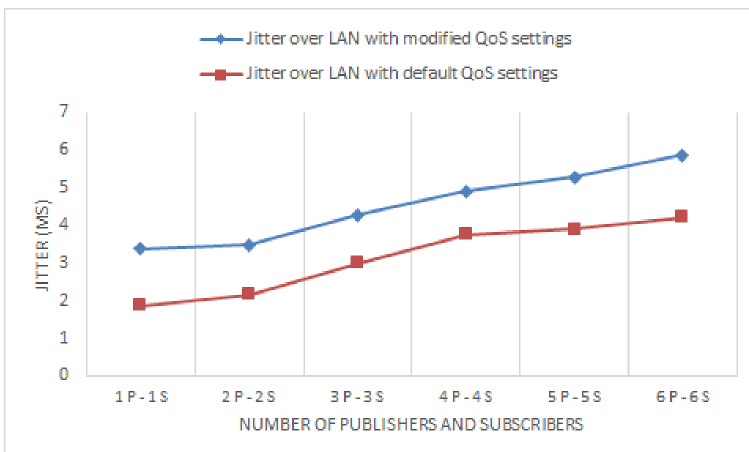

Figure 4.12: Jitter comparison over LAN with default and modified QoS settings

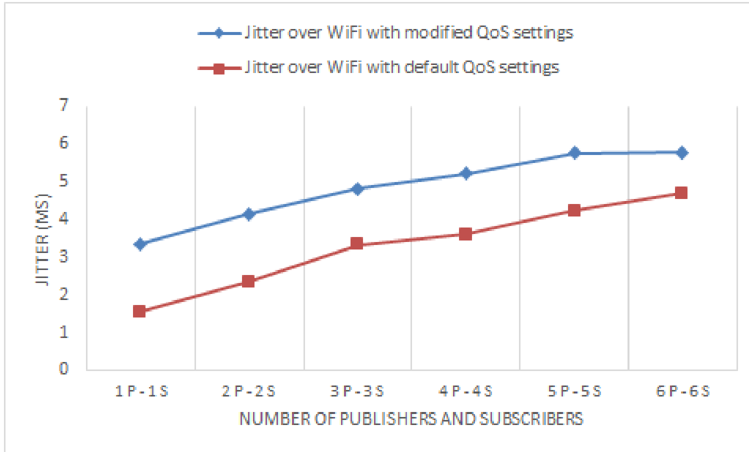

Figure 4.13: Jitter comparison over WiFi with default and modified QoS settings

Through above graphs we can analyze that by changing the values and settings of various QoS policies, we can modify performance efficiency of the system and can achieve desired results. As by tuning various QoS policies settings, our latency and jitter performance can be changed according to system requirements.

4.5.2 Throughput analysis

Throughput depends on packet size and number of packets sent over a communication channel. 12,000 to 110,000 packets were sent and received by publishers and subscribers respectively. Total time consumed for this transmission was recorded and throughput is calculated through equation 4.3. The experiment was conducted 10 times to get more accurate results. Tables 4.9 and 4.10 show the results obtained for throughput with default and modified QoS settings over LAN and WiFi. Throughput is also measured at the publisher side with default that is RELIABLE and modified that is BEST_EFFORT values of RELIABIL-

ITY QoS policy. It can be analyzed that throughput with modified QoS policies is higher than the default values because Data Writer of publisher writes all the time without taking in account the queue size of Data Reader of subscriber, when RELIABILITY QoS policy is set to BEST_EFFORT. Figures 4.14 and 4.15 show the throughput graphs. As we can see throughput increases with number of publishers and subscribers so it conveys a proportional behavior which is a legitimate one because throughput increases with increase in traffic.

Table 4.9: Throughput performance with default QoS settings

Scenario	**LAN**			**WiFi**		
P-S	**Throughput Min (Mbps)**	**Throughput Max (Mbps)**	**Throughput Mean (Mbps)**	**Throughput Min (Mbps)**	**Throughput Max (Mbps)**	**Throughput Mean (Mbps)**
1-1	6.31	12.43	6.91	5.62	13.01	7.43
2-2	6.52	11.81	7.52	6.28	13.56	8.19
3-3	7.72	12.51	7.91	7.34	13.91	8.88
4-4	7.95	14.71	8.61	7.92	14.37	9.71
5-5	8.89	14.61	10.73	8.72	14.83	10.29
6-6	12.49	16.10	11.91	8.99	15.91	12.19

Table 4.10: Throughput performance with modified QoS settings

Scenarios	**LAN**			**WiFi**		
P-S	**Throughput Min (Mbps)**	**Throughput Max (Mbps)**	**Throughput Mean (Mbps)**	**Throughput Min (Mbps)**	**Throughput Max (Mbps)**	**Throughput Mean (Mbps)**
1-1	7.63	14.72	10.34	7.52	15.72	13.62
2-2	8.67	15.89	12.63	8.62	17.92	15.05
3-3	11.82	17.25	14.82	10.62	19.52	16.78
4-4	12.52	18.83	14.58	11.53	19.45	15.72
5-5	14.82	19.76	17.26	12.83	21.62	18.53
6-6	14.09	19.93	16.99	13.89	22.61	20.17

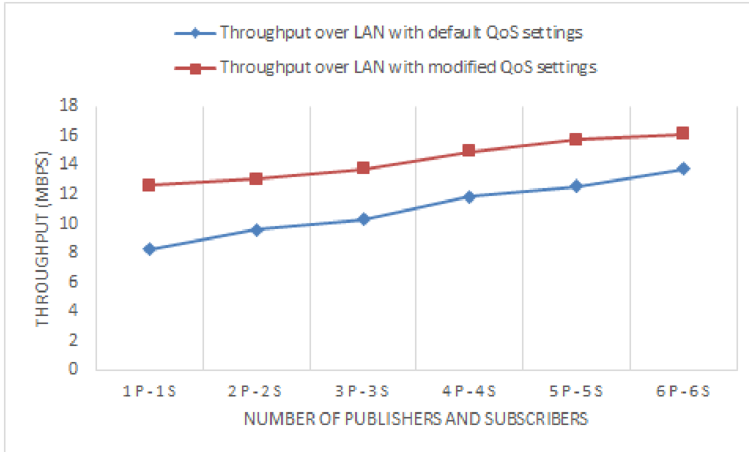

Figure 4.14: Throughput comparison over LAN with default and modified QoS settings

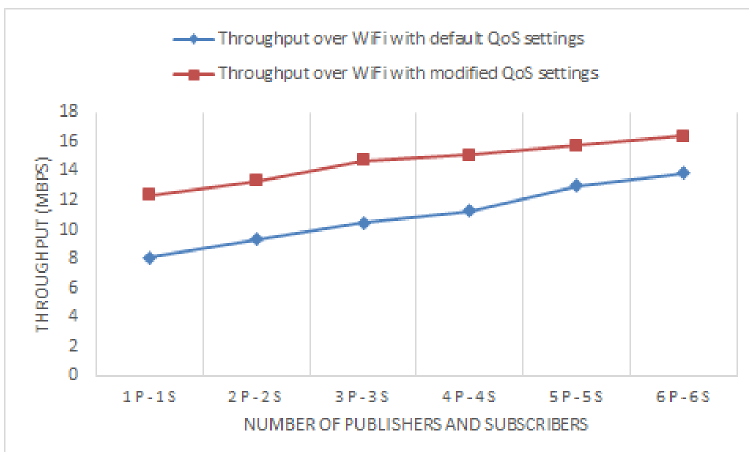

Figure 4.15: Throughput comparison over WiFi with default and modified QoS settings

Simulation graphs describes the over all behavior of the system where throughput increases tremendously by changing the values and settings of various QoS policies, we can change and increase performance efficiency of the system and achieve our desired results based on certain QoS settings. As by doing certain modifications in QoS policies settings mentioned in Table 4.5, our throughput performance has increased tremendously both over LAN and WiFi.

For each experiment, extra network traffic is generated keeping in view the heavily loaded Smart grid network and then the results are obtained. The number of publishers and subscribers are ranged from 1 to 6 because our case study depends upon six publishers and six subscribers in total, where performance measurement efficiency slightly decreases with these many participants. It is expected that latency and jitter will remain stable with increase in publishers and subscribers with default topic size of 65KB while this behavior may change with topic sizes of 256KB or more.

As Smart grid devices need a reliable communication platform that can also provide interoperability. DDS is much useful to implement such devices in Smart grid communication infrastructure. DDS offers rich sets of QoS policies that can be used for flexible and definite outcomes. For ANSI C12.19, communication experiments of end devices are performed over LAN and WiFi to prove real time feature of our proposed framework. Experimental results of latency, jitter and throughput show that this middleware can withstand with tight communication requirements while providing low latency and jitter, consuming less bandwidth and maximizing average throughput with zero packet loss or error. However it can be analyzed that latency and jitter is proportional to the number of domain participants due to increase in traffic. These results are encouraging to deploy DDS for Smart grid devices communication.

CHAPTER 5

DATA INTEGRATION OF SUSTAINABLE ENERGY RENEWABLES IN SMART GRID USING DDS

Integration of sustainable energy renewables is must in Smart grid. These renewables have much different characteristics and properties in terms of manufacturing, operation, control and data management. So they exhibit large amount of heterogeneity and are difficult to integrate with Smart grid at the same time. Sustainable energy renewables include Wind Turbine, Solar Energy, Wave Power, Hydro Electricity, Geothermal energy, Bio energy and Tidal Power etc. It is essential to control and measure the operational behavior of these renewables in order to get maximum efficiency for which real time data monitoring and analysis is necessary to make subtle decisions. We have introduced RTI Connext DDS middleware for real-time data monitoring and control of energy renewables in Smart grid. All of these systems has a controller or embedded system which will publish and subscribe data to a control station in Smart grid. Figure 5.1 describes an overview

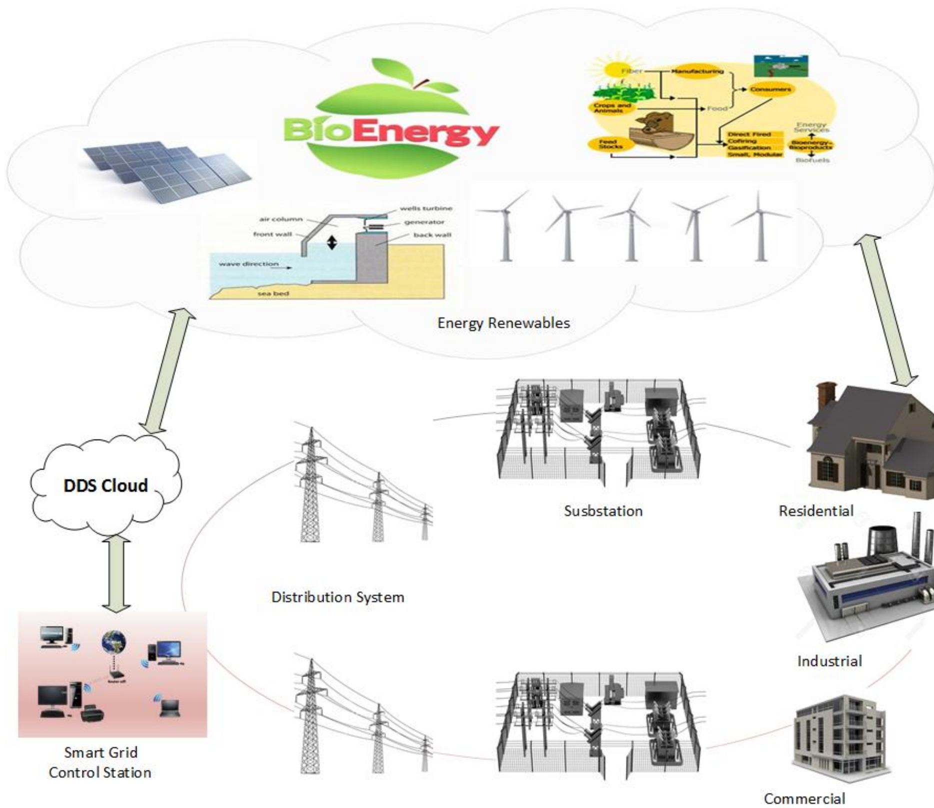

Figure 5.1: Overview of Energy renewables integration in Smart Grid based on RTPS Data Distribution Service (DDS)

of integration of energy renewables in Smart Grid based on Real Time Publish Subscribe Data Distribution Service (DDS) in which energy renewables and Smart Grid control unit communicate with each other via DDS middleware. The control unit is responsible for command and control of all the devices in Smart grid so it makes decisions based on data collected through renewables via DDS and transmits this data over distribution system to other devices if necessary. Similarly consumers may also generate their own electricity and send back to grid via two way communication channel based on DDS.

5.1 Energy renewables implementation over DDS

Figure 5.1 shows the actual implementation of real time publish subscribe DDS based application of energy renewables. Smart grid control unit and energy renewables are domain participants that communicate with each other being in the same network. Separate topics have been formed for each renewable and in this scenario, energy renewables are acting as publishers to publish the data towards Smart grid control unit. Each publisher has data writers that write the data on topics. On the other hand Smart grid control unit is acting like a subscriber that has data readers to read the data from topics. In this way energy renewables and Smart grid control unit communicate through topics that reside in global data space.

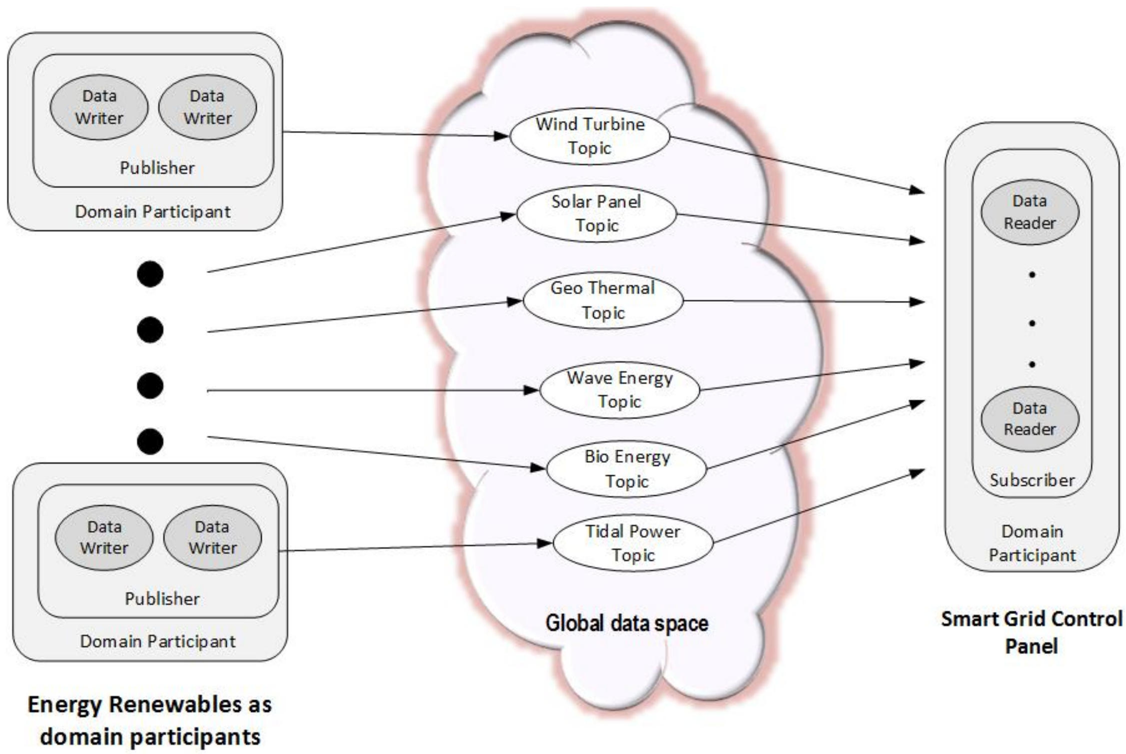

Figure 5.2: Communication methodology over DDS domain

For our experimental work setup we have construct our IDL file using six topics that are shown in Figures 5.3 to 5.8 respectively. The IDL file is generated through data structures of data control parameters of Wind Turbine [58] [59], Solar Photo Voltaic (PV) cell [60] [61], Geothermal energy [62] [63], Wave Energy [64] [65], Bio energy [66] [67] and Tidal Power [68] [69] obtained from different vendors, installers and literature work. We have develop a methodology so far based on our IDL file to establish communication among publish subscribe energy renewable applications or devices. When there is a need to transmit some data, renewables will deliver these data structures to data writer. The data writer analyzes the data whether it is according to structure of specified topic, if so then data writer writes the data on related topic. Now its time for publisher to publish the data over a communication channel.

```
struct windTurbine {
    long id; //@key
    float capacity;
    float total_cumulative_energy;
    string region_location;
    string developer;
    string date;
    string time;
    short running_hours;
    short online_hours;
    short in_rotation_hours;
    float average_power_output;
    float power_being_generated;
    float voltages;
    float current;
    float average_wind_speed;
    float wind_speed;
    float generator_speed;
    string air_direction;
    string turbine_status;
};
```

Figure 5.3: Wind Turbine IDL definition

```
struct solarPanel {

    long id; //@key

    float current;
    float reverse_saturation_current;
    float electron_charge;
    float Voltage;
    float series_resistance;
    float parallel_resistance;
    float ideality_factor;
    float boltzmanns_constant;
    float emperature;
    float shunt_resistance;
    float light_generated_current;
    float short_circuit_current;
    float open_circuit_voltage;
    float maximum_power_point;
    float fill_factor;
    float efficiency;
    float maximum_reverse_saturation_current;
    float energy_band_gap;
    float maximum_voltage;
    float maximum_current;
    float input_power;
    float light_intensity_or_irradiance;
    float surface_area_of_silicon_solar_cell;

};
```

Figure 5.4: Solar PV IDL definition

```
struct geoThermal {

    long id; //@key

    float well_depth;
    float water_temperature;
    float flow_rates;
    float pressure;
    float installed_thermal_capacity;
    float installed_power_capacity;
    float installed_heat_capacity;
    float production_Power;
    float production_heat;
    float well_head_temperature;
    float voltage;
    float current;
    float frequency;
    float specific_enthalpy;
    float absolute_pressure;
    float total_heat_duty_in_evaporator;
    float specific_heat_flow;
    float entropy;
    string time;
    float specific_volume_at_liquid_state;
    float total_work;
    float specific_actual_work;
    float specific_isentropic_work;
    float efficiency;

};
```

Figure 5.5: Geothermal energy IDL definition

```
struct waveEnergy {

    long id; //@key

    float spectral_bandwidth;
    float Wave_to_wave_correlation;
    float wave_velocity;
    float group_velocity;
    float distance_among_bodies;
    float Water_depth;
    float buoy_draft;
    float buoy_waterline_diameter;
    long bending_stiffness;
    float Wave_frequency;
    float damping_force;
    float friction_force;
    float gravity_force;
    float gravitational_acceleration;
    float drop_heigh;
    short Wave_number;
    float jet_height;
    float wave_length;
    float wave_height;
    float pressure;
    float average_power_absorption;
    float radius_of_hemisphere;
    float force_amplitude_spectrum;
    float mass_density_of_fluid;

};
```

Figure 5.6: Wave energy IDL definition

```
struct bioEnergy {

    long id; //@key

    float bio_gas_volume;
    float biogas_temperature;
    float reactor_temperature;
    float fermentation_temperature;
    float heat_input;
    float heat_natural;
    string biogas_composition;
    float biomass_consumption;
    float electricity_input;
    float produced_electricity;
    float parasitic_electricity_demand;
    float mass_of_input_liquid_feed_stocks;
    float mass_of_input_solid_feed_stocks;
    float produced_heat;
    float digester_filling_level;
    string new_feedstocks_characterization;
    string biogas_quality;
    float process_energy_efficiency;

};
```

Figure 5.7: Bioenergy IDL definition

```
struct tidalPower {

    long id; //@key

    float tecs_area;
    float channel_area;
    float availability;
    float rotor_diameter;
    short time_occurrence;
    float highest_Tide;
    float lowest_Tide;
    float largest_tidal_range;
    float smallest_tidal_range;
    float rms_water_elevation;
    short index_for_velocity_bin_numbers;
    short index_for_number_of_time_intervals;
    short number_of_time_intervals;
    short number_of_velocity;
    short number_of_data_points;
    short number_of_tecs_in_tidal_farm;
    float electrical_output_power;
    float average_electrical_power_output;
    float mean_electrical_power;
    float energy_period;
    float monochromatic_period_of_acwave;
    float magnitude_of_tidal_current;
    float elevation_above_seabed;
    float density_of_water;

};
```

Figure 5.8: Tidal power IDL definition

Publisher will publish data based on QoSs it offered. The performance of data might be different based on certain QoS parameters settings. Similarly on the other side there will be a subscriber waiting for the data that also has certain requested set of QoS parameters. QoS parameters setting on both ends is very subtle and difficult because some times QoS settings may contradict with each other that may result in incorrect outcomes that are very crucial in real time distributed applications. In this case both the communicating ends deals with Smart grid real time distributed system so one must be careful for QoS settings at each deployments. Subscriber will subscribe data from topic inserted by the publisher. Once the data is inserted in the topic by energy renewables, Smart grid control panel that requires this data will read this data through Data Reader. The data structures delivered and obtained will be same instead of just one or two fields for example time of delivery and time of received. The Smart grid control unit will now get data for further processing.

5.2 Energy renewables integration scenarios

Based on DDS components and architecture, we have made different scenarios for integration of energy renewables in Smart grid. In these scenarios, there is a Smart grid control station on one side and on the other side there are energy renwables, which will keep increasing. Same communication channel is used for all scenarios.

Our first scenario depicts the integration of wind turbines into Smart grid using DDS middleware. Because there more chances for several wind turbines to be deployed so we have taken three of them for publishing the data. Scenario 1 is shown in Figure 5.9.

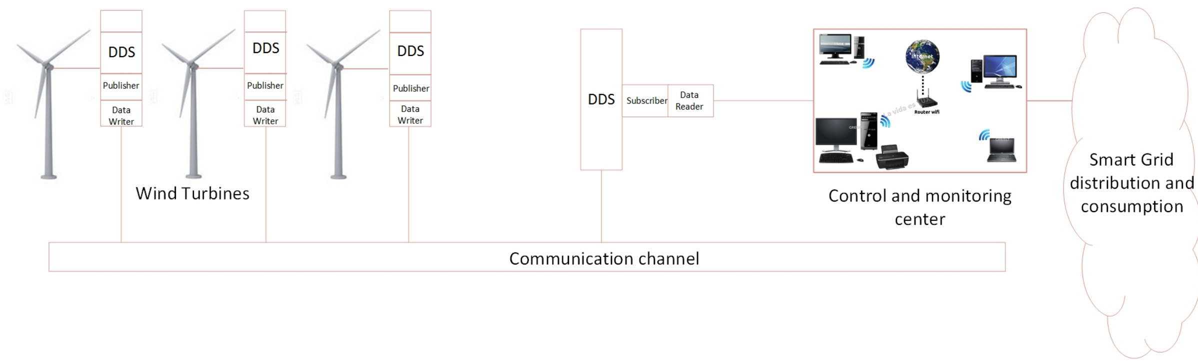

Figure 5.9: Energy renewables implementation Scenario 1

Similarly in second scenario we have added solar panels along with wind turbines to analyze the communication behavior. Scenario two is depicted in Figure 5.10 while other scenarios are depicted in Figures 5.11, 5.12, 5.13 and 5.14 where geothermal energy, wave energy, tidal power and bio-energy are depicted respectively.

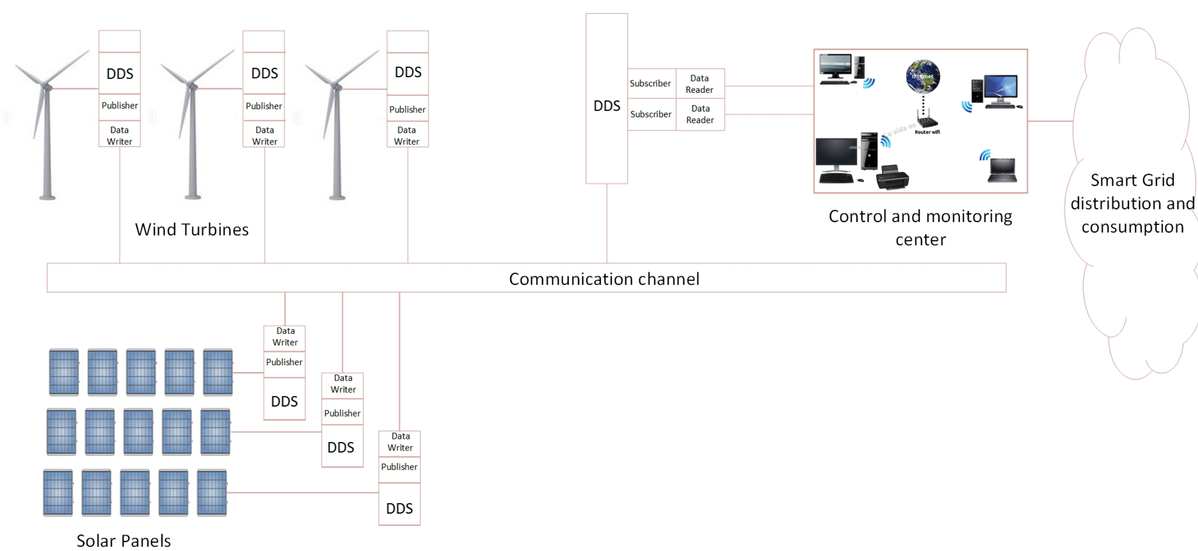

Figure 5.10: Energy renewables implementation Scenario 2

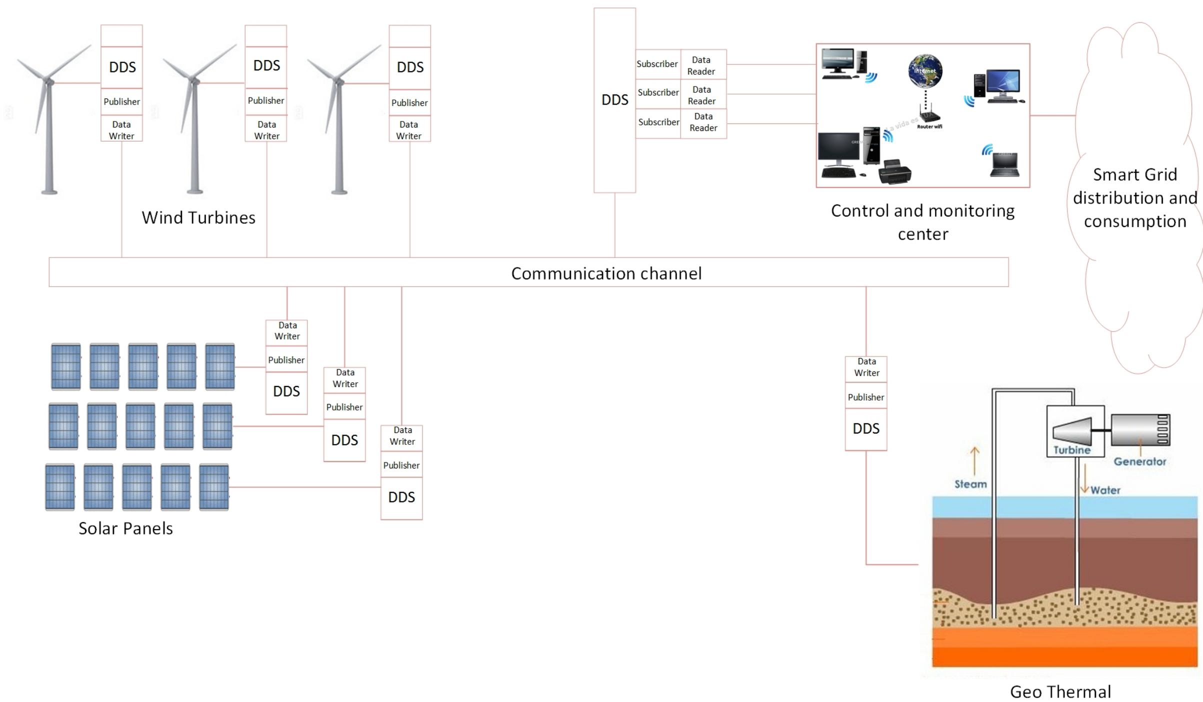

Figure 5.11: Energy renewables implementation Scenario 3

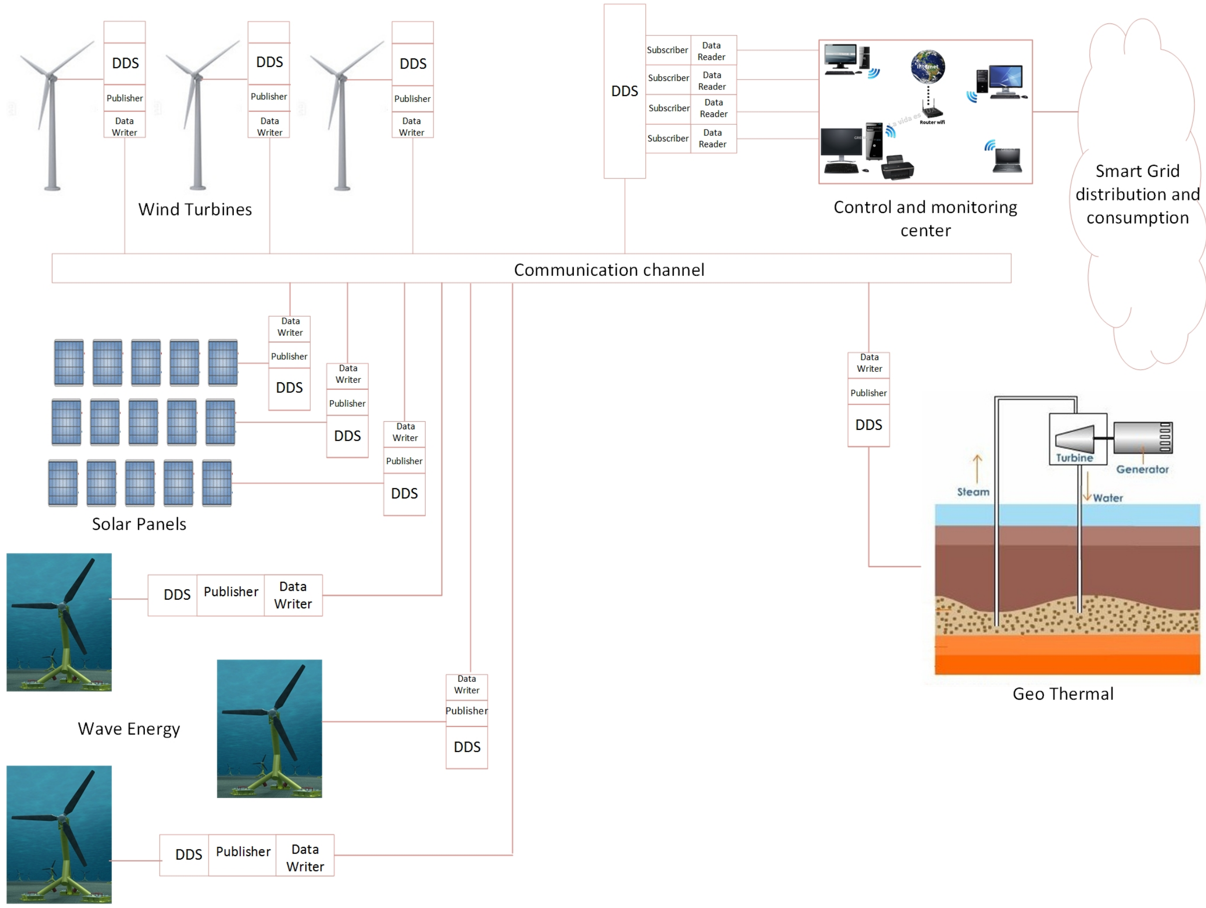

Figure 5.12: Energy renewables implementation Scenario 4

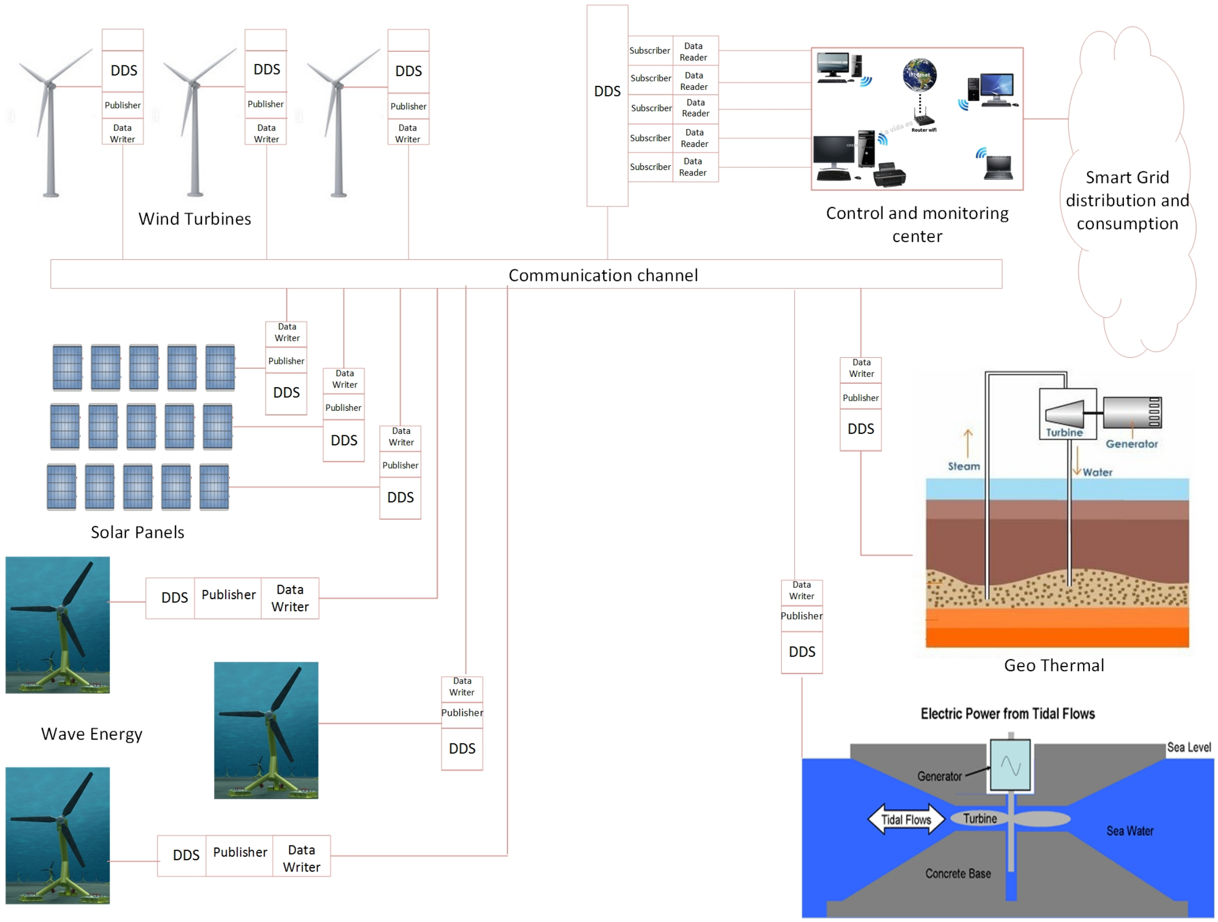

Figure 5.13: Energy renewables implementation Scenario 5

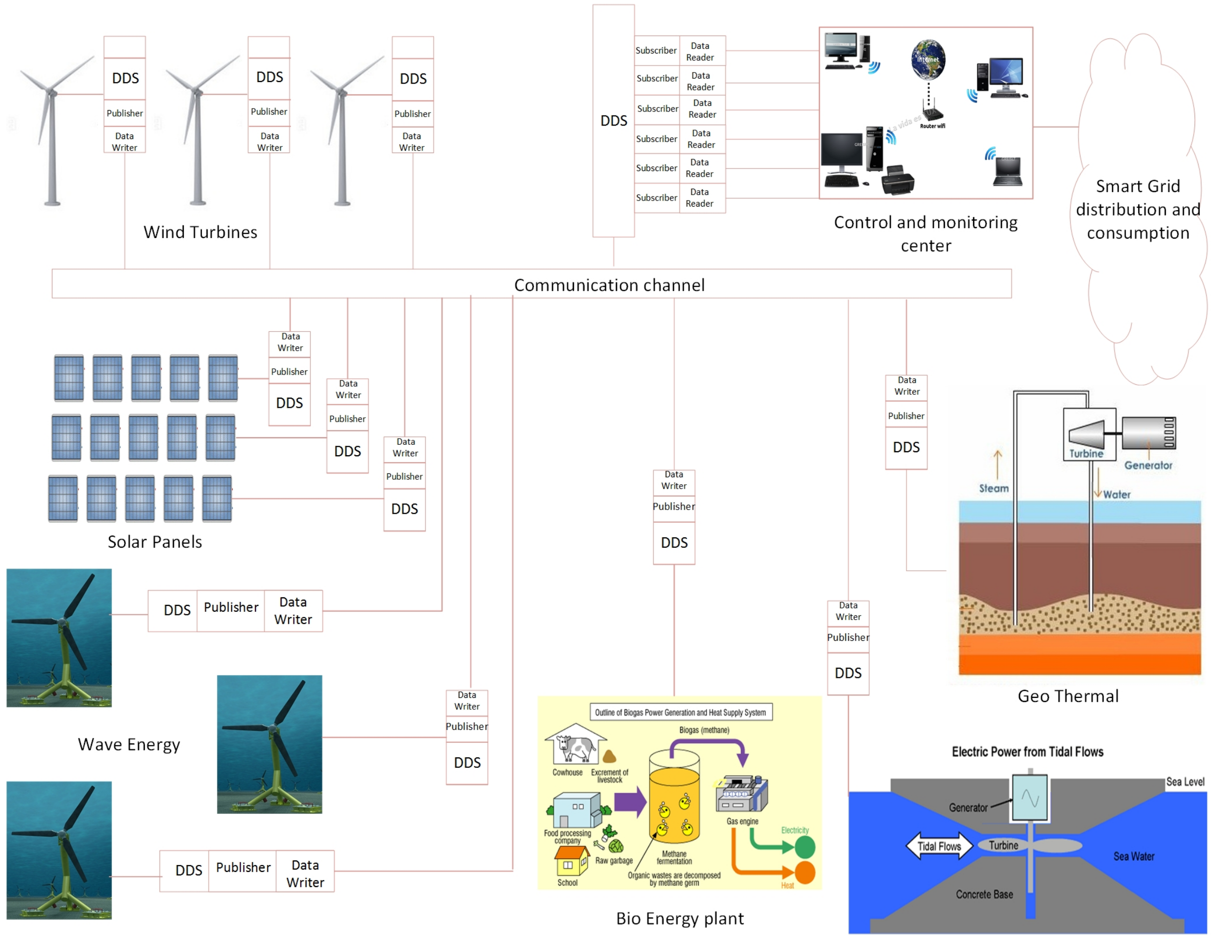

Figure 5.14: Energy renewables implementation Scenario 6

5.3 Experimental Setup

The hardware and software specifications used in experimental work is shown in Table 5.1. The experiment is performed on King Fahd University of Petroleum and Mineral's network.

Simulation behavior based on DDS is analyzed by employing many to one and many to many communication model. All the domain participants are considered to be in single domain. We have used two machines in which one is used as

Table 5.1: Hardware and Software specifications

Hosts/ Specs.	**Machine I (Publisher)**	**Machine II (Subscriber)**
CPU	Intel(R) Core 2 Duo CPU T7300 @ 2.00 GHz	Intel(R) Core i5 CPU M520 @ 2.40 GHz
OS	Windows 7 64 bit	Windows 7 32 bit
Memory	4.00 GB	2.00 GB
Network	LAN/WiFi 100 Mbps	LAN/WiFi 100 Mbps

a publisher device to publish data parameters of energy renewables and other machine as subscriber device of Smart grid control panel to subscribe that data, but these machines may also be used for both purposes at the same time. First we perform communication test by sending and receiving different packet sizes of 106, 110, 222, 226, 254, 266, 270 and 302 Bytes based on data structure sizes in various topics. RTI DDS Connext 5.2.0 is used to establish communication for both publishers and subscribers that has six topics shown in Figures 5.3 to 5.8. Extra network traffic of 10 Mbps is generated through Ostinato to load the network keeping in view the highly loaded network of Smart grid. C++ code of publishers and subscribers is generated through rtiddsgen utility and run on Visual Studio 2012 while Wireshark 1.2.3 and RTI performance test tools are used for performance measurement over LAN and WiFi.

5.4 Results and analysis

Random data values along with extra network traffic are generated at publisher's side and sent through a communication channel to one or more subscribers. The simulation is run on machines specified in Table 5.1 that are connected through LAN and WiFi. Data samples according to topic data structures are transmitted among these machines and required numeric values of all performance metrics are taken and stored for analysis.

5.4.1 Latency and jitter analysis

We used different packet sizes based on data types for our scenario to perform latency and jitter tests. Latency is measured through performance test tools provided by RTI Inc. using equation 4.1 while Jitter is measured through equation 4.2. Many to one and many to many model based experiments are performed and numeric values are taken to calculate latency and jitter. A large number of packets are sent and received among machines and tests were repeated 10 to 15 times to get more precise results. Same QoS policies are used for energy renewables experimentation as shown in Tables 4.4 and 4.5. Table 5.2 shows the results of latency with respect to default QoS policies values while Table 5.3 shows the latency with respect to modified QoS policies values. Similarly Table 5.4 shows the jitter behavior over LAN and WiFi with default and modified QoS policies values. It can be analyzed that default QoS policies values offer less latency compared to modified QoS policies.

Table 5.2: Latency Performance with default QoS policies

Scenarios	**LAN**				**WiFi**			
P-S	**Latency Min (ms)**	**Latency Max (ms)**	**Latency Average (ms)**	**Std. Dev.**	**Latency Min (ms)**	**Latency Max (ms)**	**Latency Average (ms)**	**Std. Dev.**
3-1	8.3784	14.3682	11.3678	43.02	9.3578	14.2091	11.9270	64.49
6-2	9.3781	16.5900	13.4674	48.28	10.3672	16.3894	13.8788	56.78
7-3	9.9038	17.4789	14.2789	53.29	11.4782	18.4992	15.1029	51.36
10-4	10.3789	19.6082	16.7821	56.39	11.7629	20.7738	16.3972	47.41
11-5	12.4678	21.9895	17.4890	36.48	12.4782	21.4572	17.4976	49.39
12-6	13.4801	23.8725	18.0981	50.38	13.5779	24.0018	18.3692	51.43

Table 5.3: Latency Performance with modified QoS policies

Scenarios	**LAN**				**WiFi**			
P-S	**Latency Min (ms)**	**Latency Max (ms)**	**Latency Average (ms)**	**Std. Dev.**	**Latency Min (ms)**	**Latency Max (ms)**	**Latency Average (ms)**	**Std. Dev.**
3-1	11.6359	18.3787	14.6321	34.38	11.9277	19.3651	14.8261	54.21
6-2	12.3678	19.4142	16.3269	45.29	12.4674	20.3671	17.8276	62.29
7-3	14.7849	20.6358	17.8917	56.37	14.4878	22.3681	18.7618	58.30
10-4	15.9737	22.3661	19.9971	53.02	16.3868	22.4787	20.1723	52.48
11-5	16.4679	24.1528	21.8379	41.37	16.7891	24.7582	22.3761	48.29
12-6	18.3762	26.4672	23.7369	43.47	18.3876	25.6572	24.2451	42.90

Table 5.4: Jitter performance with default and modified QoS policies

Scenarios	**With Default QoS policies**		**With Modified QoS policies**	
P-S	**LAN Jitter (ms)**	**WiFi Jitter (ms)**	**LAN Jitter (ms)**	**WiFi Jitter (ms)**
3-1	1.99	2.10	4.27	4.63
6-2	2.71	2.67	5.18	5.83
7-3	3.01	3.32	5.29	5.81
10-4	4.52	4.71	6.73	7.11
11-5	4.81	5.12	7.30	7.32
12-6	5.78	5.84	7.98	8.29

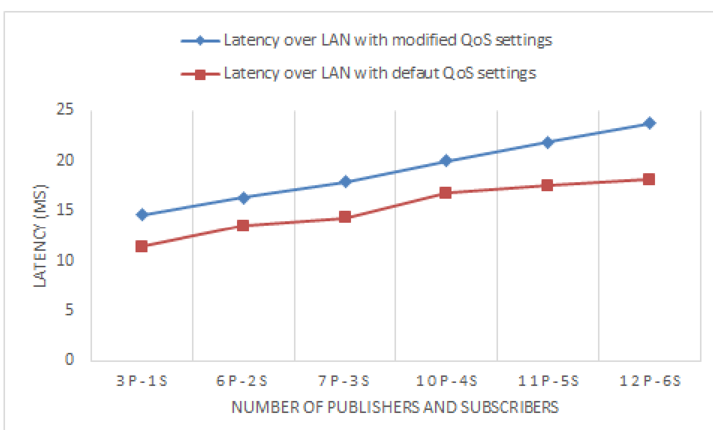

Figure 5.15: Latency comparison over LAN with default and modified QoS settings

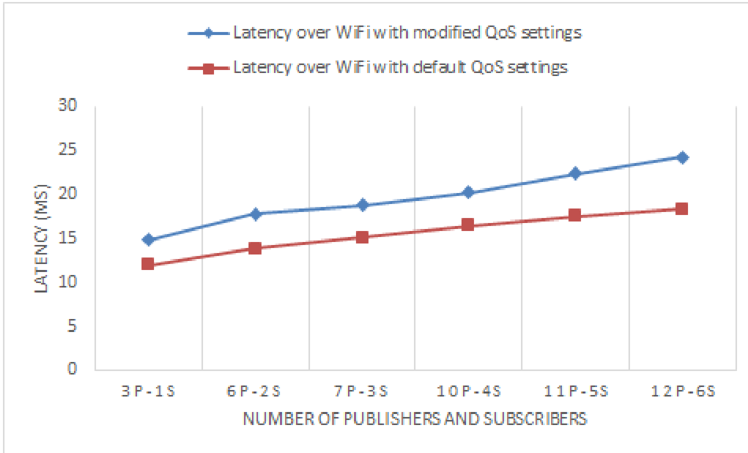

Figure 5.16: Latency comparison over WiFi with default and modified QoS settings

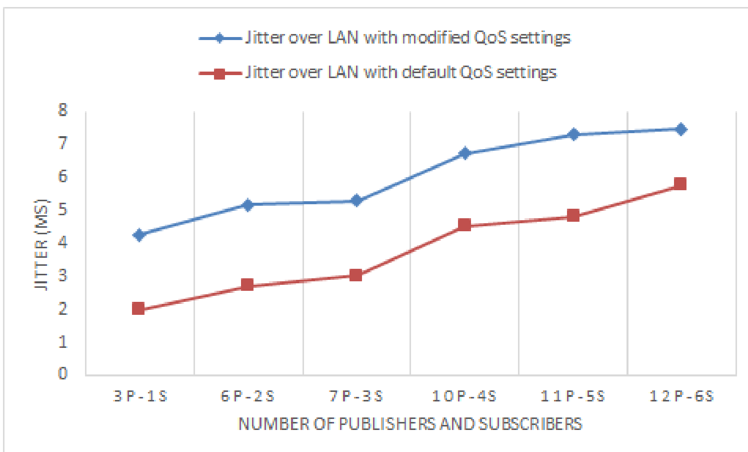

Figure 5.17: Jitter comparison over LAN with default and modified QoS settings

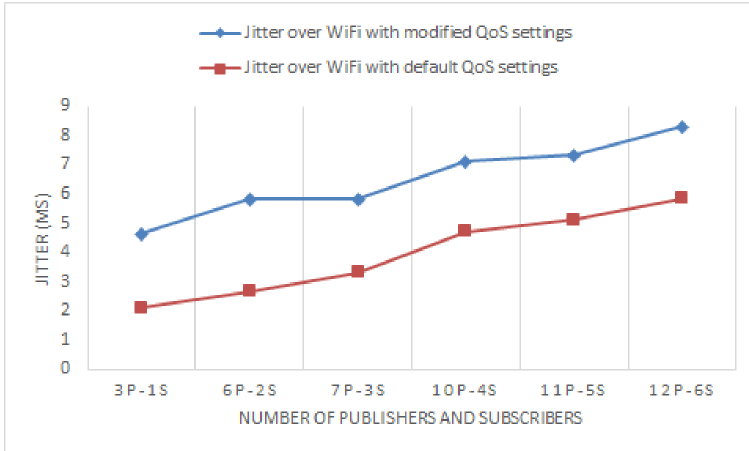

Figure 5.18: Jitter comparison over WiFi with default and modified QoS settings

Figure 5.15 shows the comparison of mean latency over LAN with default and modified QoS policies values while Figure 5.16 shows the mean latency comparison over WiFi with default and modified QoS policies. It can be analyzed that default QoS policies values offer less latency than modified QoS policies over both LAN and WiFi. This is again due to the default value RELIABLE of RELAIBILITY QoS policy, because it stops publishing data samples when Data Reader queue is full while modified value BEST_EFFORT writes all the time without noticing the queue size of Data Reader. Results are taken for both default and modified QoS settings with respect to different numbers of publishers and subscribers according to scenarios that were shown in Figures 5.9 to 5.14. Similarly jitter behavior is shown in Figures 5.17 and 5.18 where it can be seen that jitter is also less with respect to default values compared with modified QoS policies for both LAN and WiFi. All the results here show that we can change the behavior of our system by just changing the QoS settings according to requirements. The graph for each scenario can be examined where latency and jitter is almost stable with increasing number of publishers and subscribers. This behavior of Latency is much acceptable in RTPS systems where traffic increases gradually in single communication channel at a given point of time.

5.4.2 Throughput analysis

Throughput depends on packet size and number of packets sent over a communication channel. Large amount of packets were sent and received by publisher and subscriber respectively. Total time consumed for this transmission was recorded to calculate throughput. The experiment was conducted 10 times to get more accurate results. Table 5.5 shows the results obtained for throughput with respect to default QoS policies while Table 5.6 shows the throughput results with respect to modified QoS values. Throughput is measured at the publisher side with RELIABLE and BEST_EFFORT values of RELIABILITY QoS policy.

Figures 5.19 and 5.20 show the throughput graphs with default and modified QoS settings for LAN and WiFi. It can be analyzed that throughput with modified QoS policies is higher than the default values because BEST_EFFORT sends data without considering queue size as RELIABILITY does. As we can see throughput increases with number of publishers and subscribers so it conveys a proportional behavior which is a legitimate one because throughput increases with increase in traffic.

Throughout the experiments number of publishers and subscribers are increasing based on scenarios and performance measurement efficiency also decreases with these many participants. It is expected that more increase in number of participants will follow same course of measurement that has obtained in experiments of latency, jitter and throughput.

Table 5.5: Throughput performance with default QoS policies

Scenarios	**LAN**			**WiFi**		
P-S	**Throughput Min (Mbps)**	**Throughput Max (Mbps)**	**Throughput Average (Mbps)**	**Throughput Min (Mbps)**	**Throughput Max (Mbps)**	**Throughput Average (Mbps)**
3-1	9.59	14.56	11.38	10.18	15.39	11.26
6-2	10.47	16.49	13.46	10.24	26.83	13.84
7-3	11.56	18.82	16.28	11.43	18.53	15.98
10-4	12.83	19.04	17.36	12.72	19.63	17.01
11-5	15.84	12.48	19.94	12.72	22.7	19.99
12-6	17.49	24.90	21.36	18.01	24.48	21.65

Table 5.6: Throughput performance with modified QoS policies

Scenarios	**LAN**			**WiFi**		
P-S	**Throughput Min (Mbps)**	**Throughput Max (Mbps)**	**Throughput Average (Mbps)**	**Throughput Min (Mbps)**	**Throughput Max (Mbps)**	**Throughput Average (Mbps)**
3-1	11.96	18.36	14.29	11.01	18.28	14.10
6-2	13.79	20.50	16.37	12.03	21.46	16.68
7-3	14.67	13.79	18.58	14.83	13.01	19.04
10-4	16.36	25.96	20.92	16.99	24.91	20.56
11-5	17.62	26.62	23.58	18.26	25.38	23.91
12-6	18.71	28.68	24.58	18.98	27.30	24.09

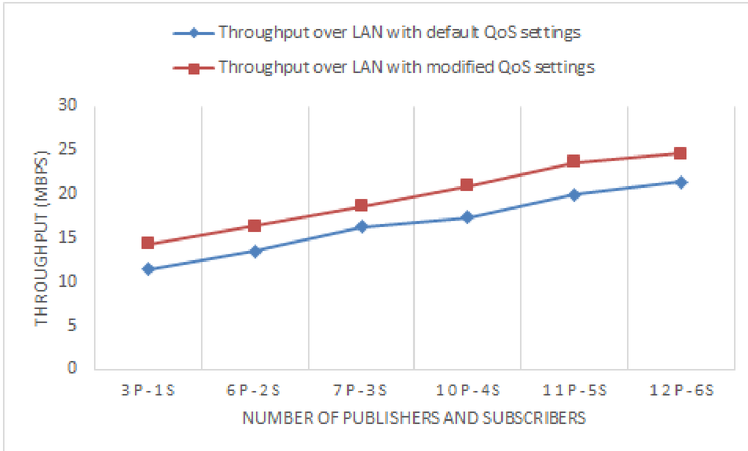

Figure 5.19: Throughput comparison over LAN with default and modified QoS settings

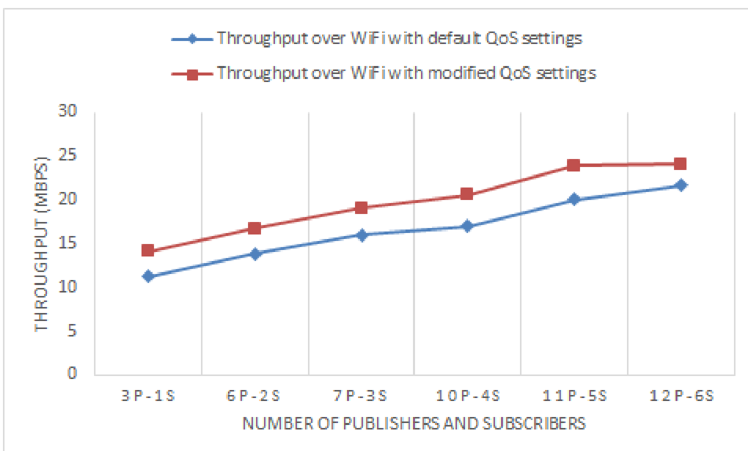

Figure 5.20: Throughput comparison over WiFi with default and modified QoS settings

Our experimental work shows that most of the distributed systems such as Smart grid includes transmission of system states and data control parameters, therefore, it needs high data rates. DDS is a middleware that can provide inter-operability and mediation especially in constrained and mission critical environments. It is a data centric communication paradigm that provides high efficiency and abstraction to distinguished data types. The results show that DDS provides such excellent communication platform for such diverse applications of integration of energy renewables in Smart grid. They propose low latency high throughput and efficient bandwidth utilization for limited data sizes. It has seen from literature study that very few attention is provided on DDS based middleware for distributed real time mission critical applications such as Smart grid. Here performing some networking experimental work we have concluded that DDS based middleware can withstand diverse communication requirements of real time and mission critical applications such as Smart grid. DDS solves the problems for integration of diverse applications and systems such as energy renewables in Smart grid by providing a complete communication infrastructure.

5.5 Socket IO Client Server model based implementation

Socket IO is a web socket API based client server model that is used for asynchronous bidirectional real time communication. Single TCP socket is used during communication either through secure or unsecure protocol that is 'ws' and 'wss' respectively from client and server. Socket applications may be executed in Safari, Opera, Chrome and Firefox. There are many applications that can be made using Socket IO with high efficiency and performance such as real time analytics, binary streaming, instant chat and messaging, document collaboration and so on [18]. Figure 5.21 depicts our communication model of energy renewables in Smart Grid in which server sends the values of energy renewables based on the requests of clients either via LAN or WiFi.

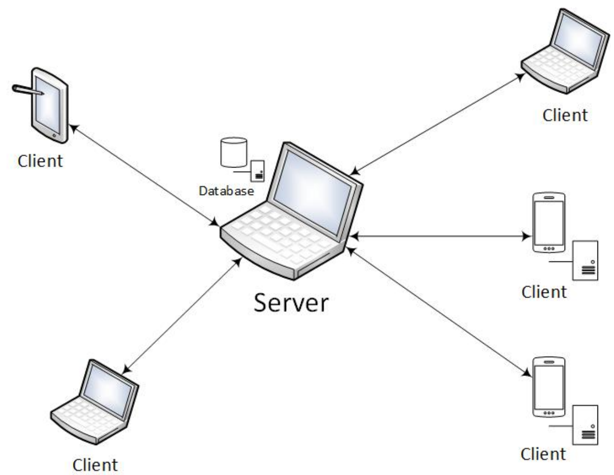

Figure 5.21: SocketIO based client server communication model

5.6 Comparison of SocketIO with DDS

Experimental work related to SocketIO web application is conducted using additional android devices (clients) as shown in Figure 5.21 on Node.js 5.0.0 using JavaScript language. Similar structures were made as in IDLs for energy renewables in data base of server and random values were sent to clients on per their requests.

5.6.1 Results and analysis

For SocketIO, random values are generated at server side and sent to clients on per their requests. Server based simulation ran on machines specified in Table 5.1 that were connected through LAN and WiFi. The generated values are transmitted among these machines and required numeric values are taken and stored through Wireshark for analysis.

5.6.2 Latency analysis

We have used different packet sizes for our scenario to perform latency tests. For DDS, default QoS settings are used to make comparison with SocketIO based client server model. One to many model based experiments are performed and numeric values are taken to calculate latency. One to many model for DDS is adopted due to client server one to many communication model, to obtain exact comparable results for both models. 12,000 to 60,000 packets are sent and received among machines and tests were repeated several times to get more precise results. Tables 5.7 and 5.8 show the results of latency of SocketIO based client server model and publish subscribe based DDS model respectively. Figures 5.22 and 5.23 shows the average latency graph over LAN and WiFi with respect to one to many models. The graph for each scenario can be examined where latency in SocketIO based client server model is approximately two times higher than DDS publish subscribe model.

Table 5.7: Latency performance in SocketIO model

Scenarios	LAN			WiFi		
P-S	Latency Min (ms)	Latency Max (ms)	Latency Average (ms)	Latency Min (ms)	Latency Max (ms)	Latency Average (ms)
1-1	17.3674	25.4671	21.5628	16.9487	25.0390	20.8726
1-2	17.5628	26.3885	21.9910	17.4663	25.8009	21.8911
1-4	18.4785	27.9573	22.8193	18.5622	26.4591	22.8583
1-6	18.9836	28.7624	23.5618	19.7738	27.8693	23.4589
1-8	19.4591	29.8577	24.3884	20.3370	29.5882	24.5782
1-10	20.6573	29.8989	25.3672	21.4982	29.7887	25.0284

Table 5.8: Latency performance in DDS model

Scenarios	LAN			WiFi		
P-S	Latency Min (ms)	Latency Max (ms)	Latency Average (ms)	Latency Min (ms)	Latency Max (ms)	Latency Average (ms)
1-1	5.4785	14.5783	9.3771	5.2763	14.2291	9.9163
1-2	5.9090	15.6833	10.3887	6.4884	15.6982	10.4820
1-4	6.3895	16.7863	11.8917	7.8710	16.3516	11.7509
1-6	7.0948	16.6990	12.7800	8.5791	17.4992	12.5629
1-8	8.9689	17.8732	13.8173	8.8778	18.5903	13.5290
1-10	9.4872	19.5886	14.8748	9.4567	19.7043	14.1521

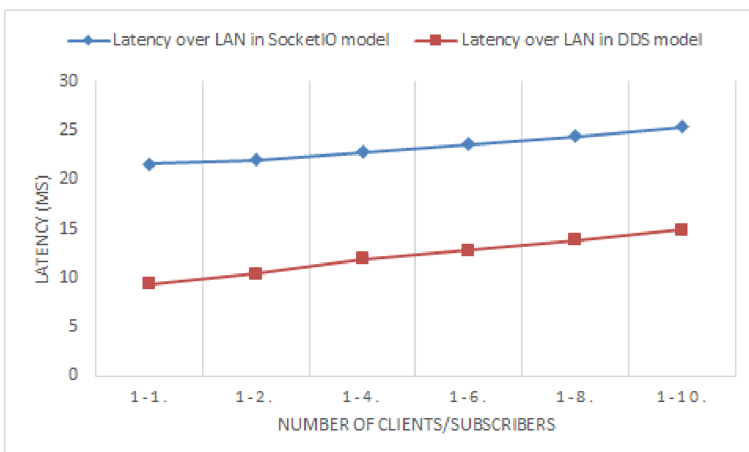

Figure 5.22: Latency analysis between Client Server and DDS over LAN

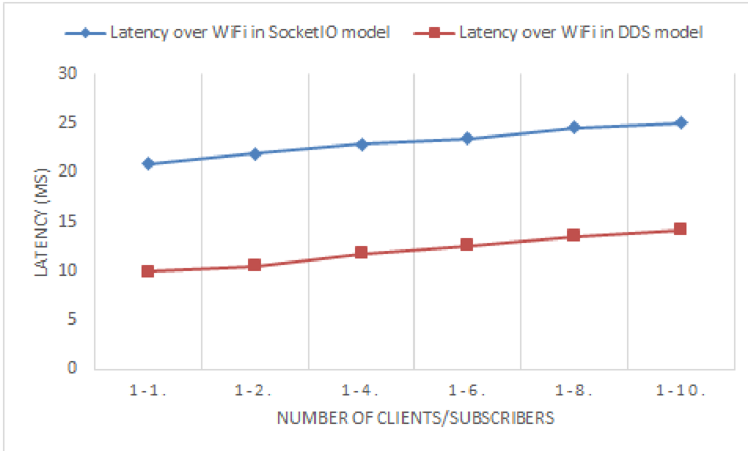

Figure 5.23: Latency analysis between Client Server and DDS over WiFi

In experiments number of subscribers and clients are ranged from 1 to 10 while number of publisher is only 1. This is design is kept to make one to one correspondence among both communication models. It can be analyzed that, with random inclusion of domain participants the performance measurement efficiency is almost stable even with these many participants. It is expected that more increase in number of subscribers or clients will follow same course of measurement that has obtained in experiments of latency. In the end we have compared the latency results of SocketIO and DDS model with already published results of Common Object Request Broker Architecture (CORBA) and Simple Object Access Protocol (SOAP) based client server model. Murlitharan Krishnan, et al. [74] have implemented Web services based Home Area Management system in Smart grid. They have measured the latency performance by establishing communication between consumer, home appliances and service providers in Smart grid. As authors have obtained their results on LAN so we have also compared our LAN results of

SocketIO and DDS based models with SOAP and CORBA to make more accurate comparison.

Table 5.9: Latency comparison between different communication models

Plot	SOAP Average Latency (ms)	CORBA Average Latency (ms)	SocketIO Average Latency (ms)	DDS Average Latency (ms)
1-1	52	48	21.5628	9.3771
1-2	53	44	21.9910	10.3887
1-3	53	46	22.1287	11.2573
1-4	57	50	22.8193	11.8917
1-5	51	48	23.3781	12.4781
1-6	55	49	23.5618	12.7800
1-7	50	50	23.8819	13.6482
1-8	58	49	24.3884	13.8173
1-9	57	48	24.8891	14.3782
1-10	53	50	25.3672	14.8748

Table 5.9 shows the latency results of SOAP, CORBA, SocketIO and DDS based communication models. All of the models used 1 Server and 1 to 10 Clients while 1 publisher and 1 to 10 Subscribers are used in DDS model to provide one to one correspondence with other models.

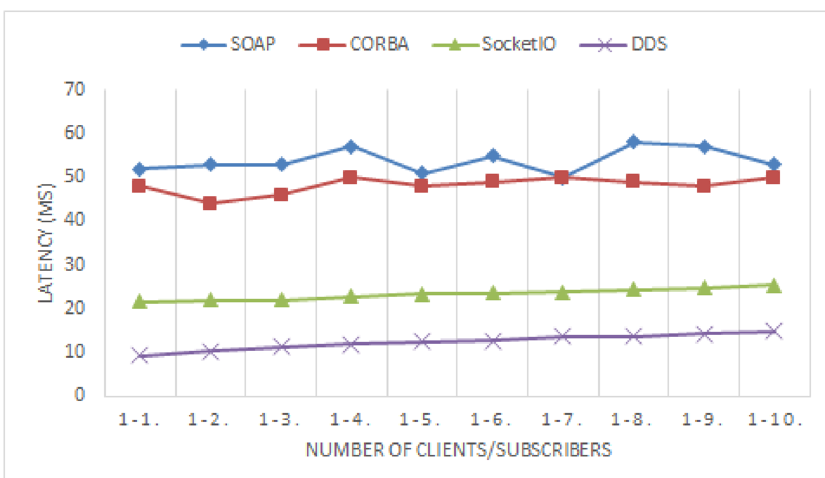

Figure 5.24: Latency analysis between CORBA, SOAP, SocketIO and DDS

Figure 5.24 shows the graph of latency comparison of all the models. Latency results of SOAP and CORBA based communication models are obtained through already published results while results of SocketIO and DDS based communication models are calculated. As SocketIO client server model is Node.js based state of the art technology so it outperforms SOAP and CORBA based models while DDS outperforms all of the models including SocketIO in terms of minimum latency.

CHAPTER 6

CONCLUSION AND FUTURE DIRECTIONS

As we deal with so many limitations in regular existing power grid so we need to shift towards Smart grid that provides numerous functionalities on demand but has high complexity in terms of heterogeneity. It has been analyzed that in order to provide interoperability and establish communication between numerous heterogeneous devices of Smart grid a QoS providing middleware is required. Several middleware architectures for Smart grid have been proposed in literature but they only provide limited functionality. Based on experimental study it can be concluded that DDS is the most suitable middleware for Smart grid communication devices. In this work data communication experiments based on DDS using Smart grid standards and energy renewables are performed over RTI Connext. DDS offers data-centric publish subscribe paradigm to provide abstraction for complex data representations of Smart grid devices. The results show that DDS deployment over communication channel can fulfill the requirements of real time systems. It offers high throughput and low latency along with high efficiency, reliability and flexibility. Results acquired here are encouraging to make more advancements in the respected field as this technology implementation is a severe need of an hour.

For future directions, DDS is essential to be deployed in almost every field of Smart grid such as Smart meters and Smart homes. As Smart grid standards are emerging day by day to improve the over all efficiency and to standardize whole grid so DDS must be analyzed to accomodate with future coming standards. Moreover there are several different networking technologies on which DDS may be deployed and analyzed as part of Smart grid communication infrastructure.

REFERENCES

[1] Deotare, Punam, and Lalit Dole. "Overview of automation of Smart Grid network." Intelligent Systems and Control (ISCO), 2015 IEEE 9th International Conference on. IEEE, 2015.

[2] The Smart Grid - An Introduction (U.S. Department of Energy)

[3] Shailesh, SL Nalbalwar1 Jayesh D. Ruikar, and R. Sakpal. "SMART GRID: A MODERNIZATION OF EXISTING POWER GRID.".

[4] Zihan Xu. "Smart Grid: Trends in Power Market". (2010).

[5] Balducci, P., et al. The smart grid: An estimation of the energy and CO2 benefits. Pacific Northwest National Laboratory, 2010.

[6] Department of Energy, U.S. (2009) Smart grid system report. http://www.doe.energy.gov/. Cited 28 Sep 2015.

[7] Momoh, James. Smart grid: fundamentals of design and analysis. Vol. 63. John Wiley & Sons, 2012.

[8] Smart Grid – The New and Improved Power Grid: A Survey.

[9] Güngör, Vehbi C., et al. "Smart grid technologies: communication technologies and standards. " Industrial informatics, IEEE transactions on 7.4 (2011): 529-539.

[10] T. Sauter and M. Lobashov, “End-to-end communication architecture for smart grids,” IEEE Trans. Ind. Electron., vol. 58, no. 4, pp.1218–1228, Apr. 2011.

[11] Kuzlu, Murat, Manisa Pipattanasomporn, and Saifur Rahman. "Communication network requirements for major smart grid applications in HAN, NAN and WAN." Computer Networks 67 (2014): 74-88.

[12] Ancillotti, Emilio, Raffaele Bruno, and Marco Conti. "The role of communication systems in smart grids: Architectures, technical solutions and research challenges." Computer Communications 36.17 (2013): 1665-1697.

[13] Bouhafs, Faycal, Michael Mackay, and Madjid Merabti. "Links to the future: communication requirements and challenges in the smart grid." Power and Energy Magazine, IEEE 10.1 (2012): 24-32.

[14] Yan, Ye, et al. "A survey on smart grid communication infrastructures: Motivations, requirements and challenges." Communications Surveys & Tutorials, IEEE 15.1 (2013): 5-20.

[15] ETSI M2M TR 102 935, “Applicability of M2M architecture to Smart Grid Networks,” ETSI, Tech. Rep., 2012.

[16] A. Aggarwal, S. Kunta, P. K. Verma, "A proposed communications infrastructure for the smart grid," in Innovative Smart Grid Technologies (ISGT), 2010, pp. 1-5.

[17] Gungor, V. Cagri, et al. "A survey on smart grid potential applications and communication requirements." Industrial Informatics, IEEE Transactions on 9.1 (2013): 28-42.

[18] U.S. Department of Energy (DOE), Communication requirements of smart grid technologies, 2010.

[19] Moslehi, Khosrow, and Ranjit Kumar. "A reliability perspective of the smart grid." Smart Grid, IEEE Transactions on 1.1 (2010): 57-64.

[20] Deshpande, M. Madden Dr J., A. Locke, and M. Madden. "Smart choices for the smart grid: using wireless broadband for power grid network transformation."Alcatel-Lucent, technology white paper 19 (2010): 20.

[21] B. Kilbourne and K. Bender, "Spectrum for smart grid: Policy recommendations enabling current and future applications," in Proc. 1st IEEE Int. Conf. Smart Grid Commun., Oct. 4–6, 2010, pp. 578–582.

[22] GTM Research, Trends in Utility Smart Grid Communications Management.

[23] D. Popa, J. Jetcheva, N. Dejean, R. Slazar, K. Hui, J. amd Monden, Applicability statement for the routing protocol for low power and lossy networks (RPL) in AMI Networks, Internet Draft, 2011.

[24] W. Wang, Y. Xu, M. Khanna, A survey on the communication architectures in smart grid, Computer Networks 55 (2011) 3604–3629.

[25] Depuru, Soma Shekara Sreenadh Reddy, Lingfeng Wang, and Vijay Devabhaktuni. ”Smart meters for power grid: Challenges, issues, advantages and status.” Renewable and sustainable energy reviews 15.6 (2011): 2736-2742.

[26] H. Li, L. Lai, and W. Zhang, “Communication requirement for reliable and secure state estimation and control in smart grid,” IEEE Transactions on Smart Grid, vol. 2, no. 3, pp. 476–486, Sep. 2011.

[27] J. Brodsy and A. McConnell, “Jamming and Interference Induced Denial-of-Service Attacks on IEEE 802.15.4-Based Wireless Networks,” Proceedings of the SCADA Security Scientific Symposium

[28] Efthymiou, Costas, and Georgios Kalogridis. ”Smart grid privacy via anonymization of smart metering data.” Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on. IEEE, 2010.

[29] Bakker, Vincent, et al. ”On simulating the effect on the energy efficiency of smart grid technologies. ” Simulation Conference (WSC), proceedings of the 2010 winter. IEEE, 2010.

[30] M. Kezunovic, F. Xu, B. Cuka, P. Myrda, “Intelligent processing of IED data for protection engineers in the Smart Grid,” MELECON 2010 - 2010 15th IEEE Mediterranean Electro technical Conference, pp.437- 442, 26-28 April 2010.

[31] J. Wen, P. Arons, W.-H. E. Liu, “The role of Remedial Action Schemes in renewable generation integrations, ” Innovative Smart Grid Technologies (ISGT), pp.1-6, 19-21 Jan. 2010.

[32] K. Moslehi and R. Kumar, “Smart grid—A reliability perspective,” Innovative Smart Grid Technologies (ISGT), pp. 1–8, Jan. 19–21, 2010.

[33] Q. Yang, J. A. Barria, and T. C. Green, “Communication infrastructures for distributed control of power distribution networks,” IEEE Trans. Ind. Inform., vol. 7, no. 2, pp. 316–327, May 2011.

[34] Fang, Xi, et al. ”Smart grid—The new and improved power grid: A survey.”Communications Surveys & Tutorials, IEEE 14.4 (2012): 944-980.

[35] Rodríguez-Molina, Jesús, et al. ”SMArc: a proposal for a smart, semantic middleware architecture focused on Smart City energy management.” International Journal of Distributed Sensor Networks 2013 (2013).

[36] Predojev, Tatjana, et al. ”A real-time middleware platform for the smart grid.”Green Communications (OnlineGreencomm), 2014 IEEE Online Conference on. IEEE, 2014.

[37] Y. Chen, Z. Huang, P. Wong, P. Mackey, C. Allwardt, J. Ma, and F. Greitzer. 2010. ”An Advanced Decision Support Tool for Electricity Infrastructure Operations.” In Critical Infrastructure Protection IV, IFIP Advances in Information and Communication Technology, vol. 342, no. 2010, ed. T. Moore and S. Shenoi, pp. 245-260.

[38] Z. Huang, N. Zhou, F. Tuffner, Y. Chen, D. Trudnowski, W. Mittelstadt, J. Hauer, and J. Dagle. 2010. "Improving Small Signal Stability through Operating Point Adjustment." In Proceedings of the 2010 IEEE Power and Energy Society General Meeting.

[39] Zhou, Liang, and Joel JPC Rodrigues. "Service-oriented middleware for smart grid: Principle, infrastructure, and application." Communications Magazine, IEEE 51.1 (2013): 84-89.

[40] Kim, Jin Sung, and Sung Jo Kim. "An object-based middleware for home network supporting the interoperability among heterogeneous devices." 2011 IEEE International Conference on Consumer Electronics (ICCE). 2011

[41] Nordström, Lars, et al. "Information and communication system architectures for wide-area monitoring and control applications." 17th Power Systems Computation Conference (PSCC), Stockholm, Sweden, 2011.

[42] Bakken, D. "Middleware, a survey article prepared for Encyclopedia of Distributed Computing."

[43] Y.-J. Kim, J. Lee, G. Atkinson, H. Kim, and M. Thottan, "SeDAX: A Scalable, Resilient, and Secure Platform for Smart Grid Communications," IEEE Journal on Selected Areas in Communications, vol. 30, no. 6, pp. 1119–1136, July 2012.

[44] Pedro Castillejo Ruben de Diego Jos ´ e-Fern ´ an Mart ´ ´ ınez, Jesus ´ Rodr´ ıguez-Molina. Middleware architectures for the smart grid: Survey and challenges in the foreseeable future. Energies, 6(7):3593–3621, July 2013.

[45] Park, Sang Oh, Jin Sung Kim, and Sung Jo Kim. "An object-based middleware supporting efficient interoperability on a smart home network." Multimedia tools and applications 63.1 (2013): 227-246.

[46] Alkhawaja, Abdel Rahman, Luis Lino Ferreira, and Michele Albano. "Message Oriented Middleware with QoS Support for Smart Grids." INForum 2012-Conference on Embedded Systems and Real Time, 2012.

[47] Yan Liu, Jared M. Chase, and Ian Gorton. A service oriented architecture for exploring high performance distributed power models. In Chengfei Liu, Heiko Ludwig, Farouk Toumani, and Qi Yu, editors, ServiceOriented Computing, volume 7636 of Lecture Notes in Computer Science, pages 748–762. Springer Berlin Heidelberg, 2012.

[48] hin, Dong-Hoon, Shibo He, and Junshan Zhang. "Robust and cost-effective architecture design for smart grid communications: A multi-stage middleware deployment approach." INFOCOM, 2014 Proceedings IEEE. IEEE, 2014.

[49] Cui, Zhenhui, et al. "A Design of Middleware Layer for QoS in Wireless Sensor Networks." Parallel and Distributed Systems (ICPADS), 2012 IEEE 18th International Conference on. IEEE, 2012.

[50] OMG, OMG. Data Distribution Service for Real-Time Systems. Technical Report OMG Available Specification formal/15-04-10, OMG, 2015.

[51] Yang, Jinsong, et al. "Data Distribution Service for industrial automation." Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012). IEEE, 2012.

[52] An, Kyoungho, et al. "A publish/subscribe middleware for dependable and real-time resource monitoring in the cloud." Proceedings of the Workshop on Secure and Dependable Middleware for Cloud Monitoring and Management. ACM, 2012.

[53] Xiong, Ming, et al. "Evaluating the performance of publish/subscribe platforms for information management in distributed real-time and embedded systems." omgwiki. org/dds (2010).

[54] Poza, José L., Juan L. Posadas, and José E. Simó. "QoS-based middleware architecture for distributed control systems." International Symposium on Distributed Computing and Artificial Intelligence 2008 (DCAI 2008). Springer Berlin Heidelberg, 2009.

[55] Calvo Gordillo, Isidro, et al. "Designing High Performance Factory Automation Applications on Top of DDS." (2013).

[56] Alkhawaja, Abdel Rahman, et al. "Qos-enabled middleware for smart grids." (2012).

[57] Shi, Kai, Yanbing Bi, and Lin Jiang. "Middleware-based implementation of smart micro-grid monitoring using data distribution service over IP networks."Power Engineering Conference (UPEC), 2014 49th International Universities. IEEE, 2014.

[58] Bi, Y. B., et al. "Mapping of IEC 61850 to Data Distribute Service for digital substation communication." Power and Energy Society General Meeting (PES), 2013 IEEE. IEEE, 2013.

[59] https://www.nema.org/Standards/ComplimentaryDocuments/ANSI-C1219-2008-contents-and-scope.pdf

[60] Le, Thi Thieu Hoa, et al. "Timed-automata based schedulability analysis for distributed firm real-time systems: a case study." International Journal on Software Tools for Technology Transfer 15.3 (2013): 211-228.

[61] http://www.renewableuk.com/en/renewable-energy/wind-energy/uk-wind-energy-database

[62] http://www.aeoluspower-windenergy.co.uk/publicData.php

[63] Chander, Subhash, et al. "A study on photovoltaic parameters of mono-crystalline silicon solar cell with cell temperature." Energy Reports 1 (2015): 104-109.

[64] Dhass, A. D., E. Natarajan, and P. Lakshmi. "An Investigation of Temperature Effects on Solar Photovoltaic cells and Modules." International Journal of Engineering 27 (2014).

[65] Agemar, Thorsten, Josef Weber, and Rüdiger Schulz. "Deep geothermal energy production in Germany." Energies 7.7 (2014): 4397-4416.

[66] Hähnlein, Stefanie, et al. "Sustainability and policy for the thermal use of shallow geothermal energy." Energy Policy 59 (2013): 914-925.

[67] Backer, G. D. Hydrodynamic design optimization of wave energy converters consisting of heaving point absorbers, Power. Diss. Dissertation submitted to obtain the academic degree of Doctor of Civil Engineering, 2010.

[68] https://www.nodc.noaa.gov/General/wave.html

[69] Drosg, Bernhard. "Process monitoring in biogas plants." IEA Bioenergy(2013).

[70] Evald, Anders, Guilin Hu, and Morten Tony Hansen. "Technology data for advanced bioenergy fuels." Lynbgy, Denmark: FORCE Technology-Department for Biomass & Waste (2013).

[71] Insaf, Ambreen, et al. "ESTIMATION OF VARIOUS TIDAL PARAMETERS AND POSSIBILITY FOR HARNESSING TIDAL ENERGY ALONG THE SOUTHEAST COASTAL AREA OF KARACHI, PAKISTAN."

[72] Neill, Simon P., M. Reza Hashemi, and Matt J. Lewis. "The role of tidal asymmetry in characterizing the tidal energy resource of Orkney."Renewable Energy 68 (2014): 337-350.

[73] Rai, Rohit. Socket. IO Real-time Web Application Development. Packt Publishing Ltd, 2013.

[74] Krishnan, Muralitharan, and Ganesh Kumar Pugalendhi. "Web Services for Home Area Management System in the Smart Grid and it's Performance." 978-1-4799-3486 IEEE 2014.

Vitae

- Name: HASSAN ALI
- Nationality: Pakistan
- Date of Birth: 18, Feb, 1988.
- Email: *m.hassan017@hotmail.com*
- Permenant Address: Lahore, Pakistan.

Academic Background: Hassan Ali received his B.S. degree in Electrical (Computer) Engineering from COMSATS Institute of Information Technology, Islamabad, Pakistan, in 2013. He has done his M.S. degree in Computer Networks from King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia. He was in faculty of Electronics Engineering department in Govt. College of Technology, Lahore, Pakistan from September 2013 to May 2014. His research interests include Energy renewables, Smart Grid, Distributed Systems, Heterogeneous communications and Real time publish subscribe systems.